Chrome 扩展开发文档

wizardforcel

Published with GitBook



目錄

Chrome 扩展开发文档

来源：Chrome 扩展开发文档

基础文档

综述

当读完了这个综述和入门之后，就可以开始创建应用（扩展）和WebApp了。 注意：WebApp是通过应用（扩展）的方式实现的，所以除非特别声明，本页所有内容都适用于WebApp。

基本概念

一个应用（扩展）其实是压缩在一起的一组文件，包括HTML，CSS，Javascript脚本，图片文件，还有其它任何需要的文件。 应用（扩展）本质上来说就是web页面，它们可以使用所有的浏览器提供的API，从XMLHttpRequest到JSON到HTML5全都有。

应用（扩展）可以与Web页面交互，或者通过content script或cross-origin XMLHttpRequests与服务器交互。应用（扩展）还可以访问浏览器提供的内部功能，例如标签或书签等。

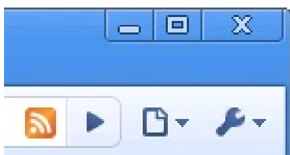
应用（扩展）的界面

很多应用（不包括WebApp）会以browser action或page action的形式在浏览器界面上展现出来。每个应用（扩展）最多可以有一个browser action或page action。当应用（扩展）的图标是否显示出来是取决于单个的页面时，应当选择page action；当其它情况时可以选择browser action。


这个gmail提醒应用使用了browser action，它在工具栏上增加一个图标



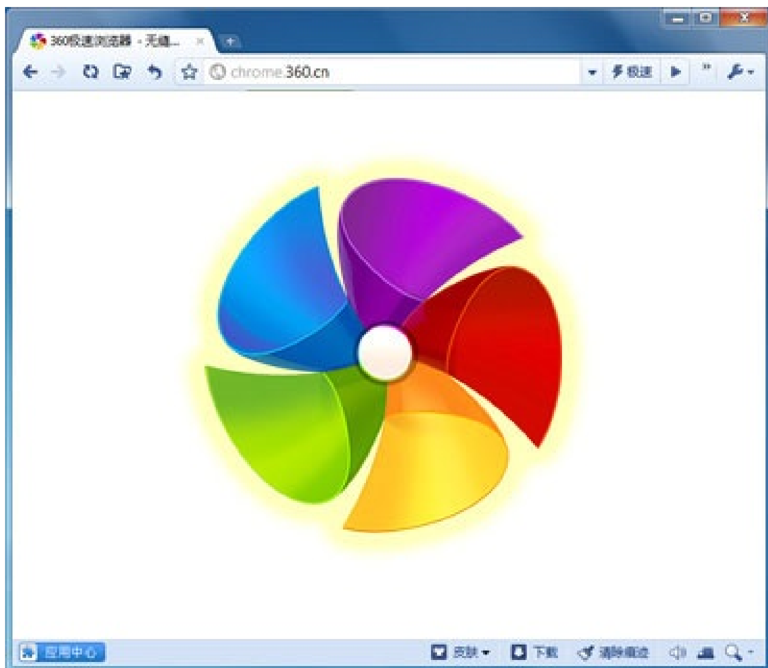
这个新闻阅读应用也使用了browser action，当点击时会弹出一个气泡窗口


这个地图应用使用了page action和content script（注入到页面内执行的脚本）

应用也可以通过其它方式提供界面，比如加入到上下文菜单，提供一个选项页面或者用一个content script改变页面的显示等。可以在"开发指南"中找到应用（扩展）特性的完整列表以及实现的细节。

WebApp界面

一个WebApp通常会打包一个包含了主要功能的html页面进来。例如下图中这个WebApp在HTML页面中显示了一个flash文件。



更多信息，查看 Packaged Apps 。

文件

每个应用（扩展）都应该包含下面的文件：

- 一个manifest文件
- 一个或多个html文件（除非这个应用是一个皮肤）
- 可选的一个或多个javascript文件
- 可选的任何需要的其他文件，例如图片

在开发应用（扩展）时，需要把这些文件都放到同一个目录下。发布应用（扩展）时，这个目录全部打包到一个应用（扩展）名是.crx的压缩文件中。如果使用Chrome Developer Dashboard,上传应用（扩展），可以自动生成.crx文件。

引用文件

任何需要的文件都可以放到应用（扩展）中，但是怎么使用它们呢？一般的说，可以像在普通的HTML文件中那样使用相对地址来引用一个文件。下面的例子演示了如何引用images子目录下的文件myimage.png

```
<img **src="images/myimage.png"**>
```

如果使用360极速版内置的调试器（开发人员工具），可以看到每一个应用（扩展）中的文件也可以用一个绝对路径来表示：

> **chrome-extension://**<*extensionID*>/<*pathToFile*>

在这个URL中，<extensionid>是为每一个应用（扩展）生成的唯一ID。从chrome://extensions页面中可以看到已经安装的所有应用（扩展）的唯一ID。<pathtofile>是文件在应用（扩展）目录下的路径，也就是它的相对路径。</pathtofile></extensionid>

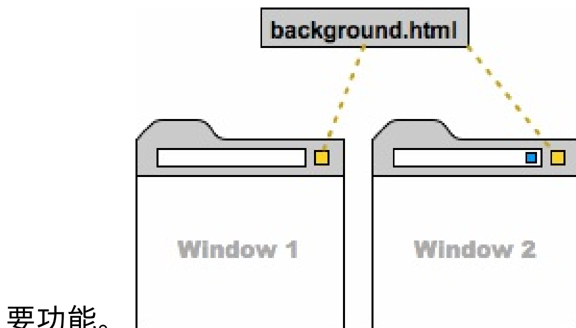
在这个URL中，<extensionid>名为manifest.json的文件包含了应用（扩展）的基本信息，例如最重要的文件列表，应用（扩展）所需要的权限等。下面是一个典型的应用（扩展），使用了browser action并访问google.com </extensionid>

{ "name": "My Extension", "version": "2.1", "description": "Gets information from Google.", "icons": { "128": "icon_128.png" }, "background_page": "bg.html", "permissions": ["http://*.google.com/", "https://*.google.com/"], "browser_action": { "default_title": "", "default_icon": "icon_19.png", "default_popup": "popup.html" } }

详细信息，参考 Manifest Files 。

基本架构

绝大多数应用（扩展）都包含一个背景页面(background page)，用来执行应用（扩展）的主要功能。



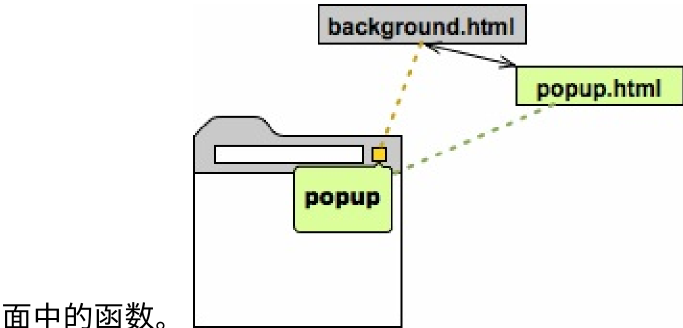
上图显示了安装了两个应用（扩展）的浏览器。两个应用（扩展）分别是黄色图标代表的browser action和蓝色图标代表的page action。在background.html文件里定义了browser action和javascript代码。在两个窗口里browser action都可以工作。

页面

背景页面并不是应用（扩展）中唯一的页面。例如，一个browser action可以包含一个弹窗(popup)，而弹窗就是用html页面实现的。应用（扩展）还可以使用chrome.tabs.create()或者window.open()来显示内部的HTML文件。

应用（扩展）里面的HTML页面可以互相访问各自DOM树中的全部元素，或者互相调用其中的函数。

下图显示了一个browser action的弹窗的架构。弹窗的内容是由HTML文件（popup.html）定义的web页面。它不必复制背景页面(background.html)里的代码,因为它可以直接调用背景页面中的函数。

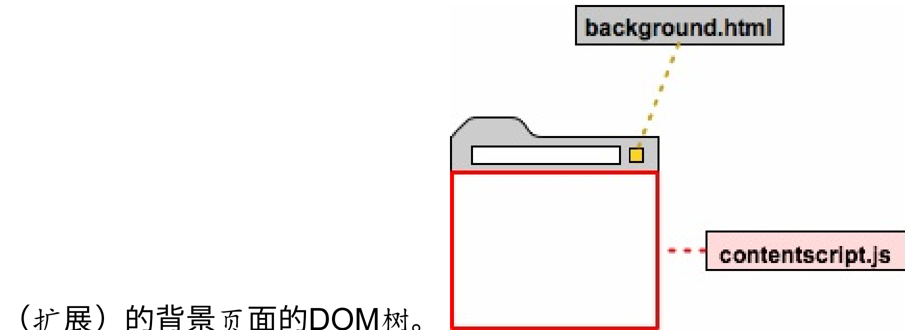


更多细节可以参考 Browser Actions 和 页面间的通信 。

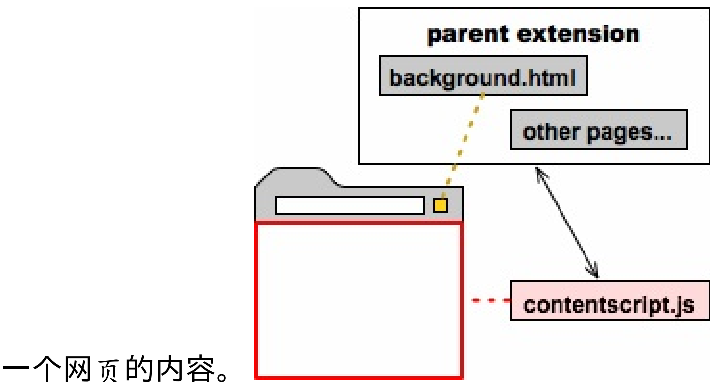
Content scripts

如果一个应用（扩展）需要与web页面交互，那么就需要使用一个content script。Content script脚本是指能够在浏览器已经加载的页面内部运行的javascript脚本。可以将content script看作是网页的一部分，而不是它所在的应用（扩展）的一部分。

Content script可以获得浏览器所访问的web页面的详细信息，并可以对页面做出修改。下图显示了一个content script可以读取并修改当前页面的DOM树。但是它并不能修改它所在应用（扩展）的背景页面的DOM树。



Content script与它所在的应用（扩展）并不是完全没有联系。一个content script脚本可以与所在的应用（扩展）交换消息，如下图所示。例如，当一个content script从页面中发现一个RSS种子时，它可以发送一条消息。或者由背景页面发送一条消息，要求content script修改一个网页的内容。



更多的信息可以查看 Content Scripts 。

页面间的通信

一个应用（扩展）中的HTML页面间经常需要互相通信。由于一个应用（扩展）的所有页面是在同一个进程的同一个线程中运行的，因此它们之间可以直接互相调用各自的函数。

可以使用 `chrome.extension` 中的方法来获取应用（扩展）中的页面，例如getViews()和getBackgroundPage()。一旦一个页面得到了对应用（扩展）中其它页面的引用，它就可以调用被引用页面中的函数，并操作被引用页面的DOM树。

保存数据和隐身模式

应用（扩展）可以使用HTML5的 Web Storage API（例如localStorage）来保存数据，或者向服务器发出请求来保存数据。当需要保存数据的时候，首先需要确定是否从隐身模式窗口中发出的请求。缺省情况下，应用（扩展）是不会运行在隐身模式下的，而webapp是会的。需要明确用户在隐身模式下究竟需要应用（扩展）或webapp做什么。

隐身模式保证在该窗口下浏览不会留下痕迹。当处理隐身窗口的数据时，一定要遵循这个前提。例如，如果一个的应用（扩展）的功能是将浏览历史保存在云端（服务器），那么不要保存隐身模式下的浏览历史。另一方面，任何窗口下都可以保存应用（扩展）的数据，不论是否隐身。

重要规则：如果一条数据可能表明用户在网上看了什么或做了什么，不要在隐身模式下保存它。

要检查窗口是否在隐身模式下，检查Tab或Window对象的incognito属性。例如：

var bgPage = chrome.extension.getBackgroundPage();

function saveTabData(tab, data) { if (tab.incognito) { bgPage[tab.url] = data; // Persist data ONLY in memory } else { localStorage[tab.url] = data; // OK to store data }

后续

现在应用（扩展）的基本知识已经介绍完了，可以开始写自己的应用（扩展）了。更多的信息可以参考：

- 入门指南
- 调试指南
- 开发指南

- 代码例子

调试

这个指南会向您介绍如何使用Chromium的内建开发工具进行应用（扩展）调试。

查看应用（扩展）信息

1.加载Hello World应用（扩展）。如果这个应用（扩展）正在运行中，你将在浏览器右边的地址栏上看到Hello World的图标。 如果这个应用（扩展）并未运行，你可以找到应用（扩展）文件，并且加载它们。如果你没有应用（扩展）文件，可以在这里下载它的zip文件，然后按创建扩展范例的方法加载它。

2.前往扩展页，地址 chrome://extensions，确认这个页面在开发者模式下。

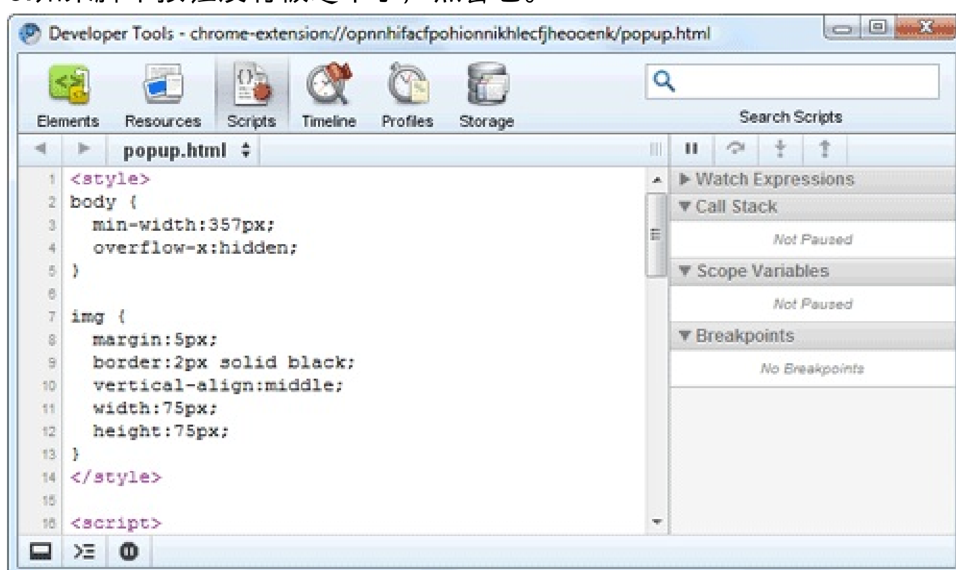
3.查看Hello World应用（扩展）的信息，你能看到应用（扩展）的名称、描述、应用（扩展）ID。

观察popup页面

1.前往扩展页，地址 chrome://extensions，确认这个页面在开发者模式下，扩展页并不需要被打开，只要浏览器记住了这个设置即可。

2.右键点击Hello World 图标 ，然后选择“审查弹出内容”菜单项，popup.html将会被显示在开发工具窗口中。

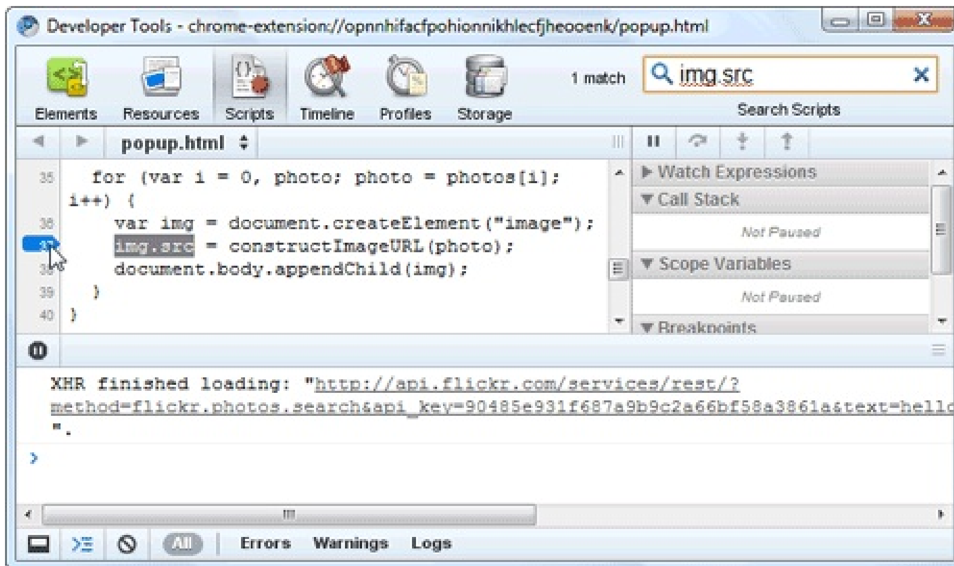
3.如果脚本按钮没有被选中了，点击它。



4.点击控制台按钮，（在开发者窗口的左下角，第二个按钮就是）这样你可以即看到代码，又看到控制台。

使用调试器

1.搜索”img.src”，然后在这个位置设置断点，只要在行号上单击即可设置。（比如：37行）。

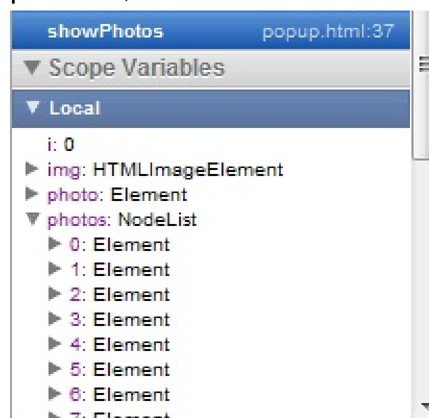


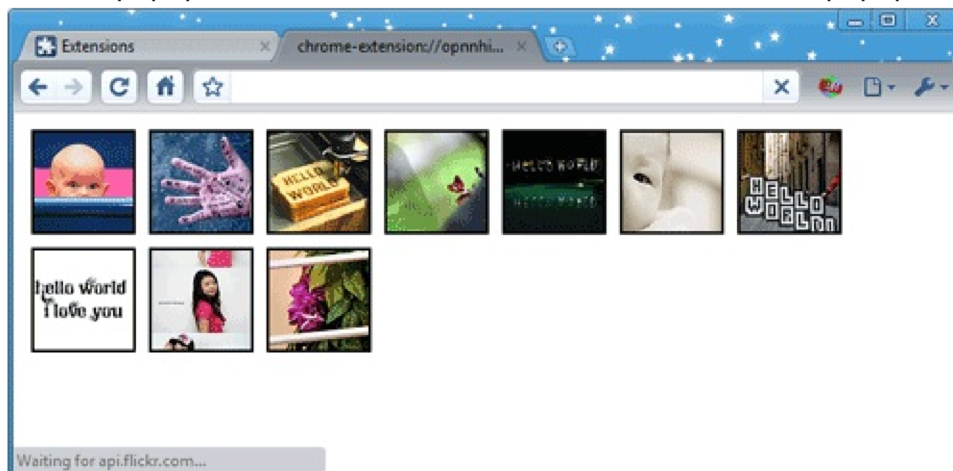
2.确认你能看到popup.html标签，它将显示20个“hello world”图片。

3.在控制台上，重新加载这个页面。命令：location.reload(true);

> location.reload(true)

4.在工具窗口的右上方，你能看到局部变量。在这个例子中，它会显示出所有当前范围所见的变量。例如：在下面的屏幕截图上，变量i=0，photos则列出了一部分的Element列表。实际上，它包含了20个元素，每个代表一个图片。



5.点击play/pause按钮（在开发者工具窗口的上右方）做一次图像处理循环，每次你点击这个按钮，i会加1，popup页面会显示更多另外的图标，当i是10的时候，popup页看起来像如下图所示：

6.点击play/pause按钮后面的的按钮，可以步进到函数调用的上层，为了让页面完成加载，点击之前的第37行，禁用之前设置的断点，接着按下play/pause按钮 继续执行。

总结

这个指南示范了简单的应用（扩展）调试，总结一下：

1.在扩展管理页面上(chrome://extensions)，找到应用（扩展）ID;

2.查看应用（扩展）中的文件，使用类似这样的格式访问 chrome-extension://extensionId/filename

3.使用开发者工具设置脚本断点，单步调试，查看变量

4.使用控制台命令 location.reload(true)来重新加载当前的调试页面

格式：Manifest文件

目录

每一个扩展、可安装的WebApp、皮肤，都有一个JSON格式的manifest文件，叫manifest.json，里面提供了重要的信息 。

字段说明

下面的JSON示例了manifest支持的字段，每个字段都有连接指向专有的说明。必须的字段只有：name和version。

```
{
_// 必须的字段_
  "[name](#name)": "_My Extension_",
  "[version](#version)": "_versionString_",
  "[manifest_version](#manifest_version)": 2,
  _// 建议提供的字段_
  "[description](#description)": "_A plain text description_",
  "[icons](#icons)": { ... },
  "[default_locale](#default_locale)": "_en_",
  _// 多选一，或者都不提供_
  "[browser_action](browserAction.html)": {...},
  "[page_action](pageAction.html)": {...},
  "[theme](themes.html)": {...},
  "[app](#app)": {...},
  _// 根据需要提供_
  "[background](background_pages.html)": {...},
  "[chrome_url_overrides](override.html)": {...},
  "[content_scripts](content_scripts.html)": [...],
  "[content_security_policy](contentSecurityPolicy.html)": "_policyString_",
  "[file_browser_handlers](http://code.google.com/chrome/extensions/fileBrowserHandler.ht
  "[homepage_url](#homepage_url)": "http://_path/to/homepage_",
  "[incognito](#incognito)": "spanning" _or_ "split",
  "[intents](#intents)": {...}
  "[key](#key)": "_publicKey_",
  "[minimum_chrome_version](#minimum_chrome_version)": "_versionString_",
  "[nacl_modules](#nacl_modules)": [...],
  "[offline_enabled](#offline_enabled)": true,
  "[omnibox](omnibox.html)": { "keyword": "_aString_" },
  "[options_page](options.html)": "_aFile_.html",
  "[permissions](#permissions_)": [...],
  "[plugins](npapi.html)": [...],
  "[requirements](#requirements)": {...},
  "[update_url](autoupdate.html)": "http://_path/to/updateInfo_.xml",
  "[web_accessible_resources](#web_accessible_resources)": [...]
}
```

字段的含义

部分字段的详细的含义如下：

app

可安装的webapp，包括打包过的app，需要这个字段来指定app需要使用的url。最重要的是app的启动页面------当用户在点击app的图标后，浏览器将导航到的地方。

更详细的信息，请参考文档hosted apps和packaged apps.：

default_locale

指定这个扩展保的缺省字符串的子目录：_lcoales。如果扩展有_locales目录，这个字段是必须的。如果没有_locales目录，这个字段是必须不存在的。具体见Internationalization.。

description

描述扩种的一段字符串（不能是html或者其他格式，不能超过132个字符）。这个描述必须对浏览器扩展的管理界面和Chrome Web Store都合适。你可以指定本地相关的字符串，具体参考:Internationalization。

homepage_url

这个扩展的主页 url。扩展的管理界面里面将有一个链接指向这个url。如果你将扩展放在自己的网站上，这个url就很有用了。如果你通过了Extensions Gallery和Chrome Web Store来分发扩展，主页 缺省就是扩展的页面。

icons

一个或者多个图标来表示扩展，app，和皮肤。你通常可以提供一个128x128的图标，这个图标将在webstore安装时候使用。扩展需要一个48x48的图标，扩展管理页面需要这个图标。同时，你还可以提供给一个16x16的图标作为扩页面的fa网页图标 。这个16x16的图标，还将显示在实验性的扩展infobar特性上。

图标要求是png格式，因为png格式是对透明支持最好的。你也可以用其他webkit支持的格式，如BMP,GIF,ICON和JPEG。下面有个例子：

```
"icons":
  {
    "16": "icon16.png",
    "48": "icon48.png",
    "128": "icon128.png"
  },
```

注意：请只使用文档说明的图标大小。

可能你已经注意到了，chrome有时候会将这些图标尺寸变小，比如，安装对话框将128-像素图标缩小为69-像素了。

然而，Chrome的界面细节可能每个版本都不一样，但每次变动都假设开发者使用的是文档标注过的尺寸。如果你使用了其他的尺寸，你的图标可能看起来很丑，在将来的某个版本中。

如果你使用Chrome Developer Dashboard上传你的扩展、app、皮肤，你需要上传附带的图片，包括至少一张扩展的缩略图。更多信息请参考 ：Chrome Web Store developer documentation。

incognito

可选值："spanning"和"split"，指定当扩展在允许隐身模式下运行时如何响应。

扩展的缺省值是Spanning，这意味着扩展将在一个共享的进程里面运行。隐身标签页的事件和消息都会发送到这个共享进程，来源通过incognito标志来区分。

可安装的webapp的缺省值是split，这个意思是隐身模式下的webapp都将运行在他们自己的隐身进程中。如果app或扩展有背景页面，也将运行在隐身进程中。隐身进程和普通进程一样，只是cookie保存在内存中而已。每个进程只可看到和自己相关的事件和消息（比如，隐身进程只能看到隐身标签也更新）。这些进程之间不能互相通信。

根据经验，如果你的扩展或app需要在隐身浏览器里面开一个标签页，使用split；如果你的扩展或app需要登记录到远程服务器或者本地永久配置，用spanning。

intents

一个字典，用于描述扩展或app所提供的全部intent handler。字典里的每个键指定了一个action verb。下面这个例子为"http://webintents.org/share"这个action verb指定了2个的intent handler。

```
{
    "name": "test",
    "version": "1",
    "intents": {
      "http://webintents.org/share": [
        {
          "type": ["text/uri-list"],
          "href": "/services/sharelink.html",
          "title" : "Sample Link Sharing Intent",
          "disposition" : "inline"
        },
        {
          "type": ["image/*"],
          "href": "/services/shareimage.html",
          "title" : "Sample Image Sharing Intent",
          "disposition" : "window"
        }
      ]
    }
  }
```

“type”指定handler所支持的一组MIME。

“href”指定了处理intent的页面URL。对于托管的apps，这个URL必须在属于允许URL集。对于扩展，这个页面必须是扩展自带的，其URL是相对扩展根目录的相对路径。

当用户触发handler对应的动作时，“title”会显示在intent选择界面中。

“disposition”可以是“inline”或“window”。当intent被触发时，“window”表示将在一个新标签中打开，而“inline”表示直接在当前标签页打开。

更多Web Intent信息请参考网站： Web Intents specification 和 webintents.org。

key

开发时为扩展指定的唯一标识值。

注意：通常您并不需要直接使用这个值，而是在您的代码中使用相对路径或者chrome.extension.getURL()得到的绝对路径。

这个值并不是开发时显式指定的，而是Chrome在安装 `.crx` 时辅助生成的。(开发时可以通过上传扩展或者手工打包生成crx文件)。 安装完crx，在Chrome的用户数据目录下的 Default/Extensions/_<extensionId>_/_<versionString>_/manifest.json 文件中，您可以看到这个扩展的key。

minimum_chrome_version

扩展，app或皮肤需要的chrome的最小版本，如果有这个需要的话。这个字符串的格式和version字段一样。

name

用来标识扩展的简短纯文本。这个文字将出现在安装对话框，扩展管理界面，和store里面。你可以指定一个本地相关的字符串为这个字段，具体参考：Internationalization。

nacl_modules

一个或多个从MIME到处理这个MIME的本地客户端模块之间的映射。 例如，下段代码中加粗部分将一个本地客户端模块注册为处理OpenOffice电子表格MIME。

```
{
    "name": "Native Client OpenOffice Spreadsheet Viewer",
    "version": "0.1",
    "description": "Open OpenOffice spreadsheets, right in your browser.",
    **"nacl_modules": [{
      "path": "OpenOfficeViewer.nmf",
      "mime_type": "application/vnd.oasis.opendocument.spreadsheet"
    }]**
  }
```

"path" 指定一个NaCl 的manifest（就像扩展有各自的manifest一样，NaCl 也有自己的manifest文件，不同的是NaCl 的以 `.nmf` 作为后缀）。 这个路径是相对于扩展根目录的。 更多NaCl 信息和 `.nmf` 文件格式请参考NaCl 技术概述。

一个MIME只能与一个".nmf"文件关联，但一个".nmf"文件可处理多个MIME。下面例子的扩展有2个".nmf"文件，但处理了3个MIME。

```
{
    "name": "Spreadsheet Viewer",
    "version": "0.1",
    "description": "Open OpenOffice and Excel spreadsheets, right in your browser.",
    "nacl_modules": [{
      "path": "OpenOfficeViewer.nmf",
      "mime_type": "application/vnd.oasis.opendocument.spreadsheet"
    },
    {
      "path": "OpenOfficeViewer.nmf",
      "mime_type": "application/vnd.oasis.opendocument.spreadsheet-template"
    },
    {
      "path": "ExcelViewer.nmf",
      "mime_type": "application/excel"
    }]
  }
```

注意：扩展不在manifest中指定“nacl_modules”也可以使用NaCl。仅在扩展希望自己的NaCl被浏览器知道并用于显示关联的MIME内容时才需要指定。

offline_enabled

指定本扩展或app是否支持脱机运行。当Chrome检测到处于脱机状态，此项设置为“是”的app将会在新标签页中高亮显示。

permissions

扩展或app将使用的一组权限。每个权限是一列已知字符串列表中的一个，如geolocatioin或者一个匹配模式，来指定可以访问的一个或者多个主机。权限可以帮助限定危险，如果你的扩展或者app被攻击。一些权限在安装之前，会告知用户，具体参考：Permission Warnings.

如果一个扩展api需要你的声明一个权限在manifest文件，一般的，api的文档将告诉怎么做。例如，Tabs页面告诉你这么声明一个tabs权限。

这是一个扩展的manifest文件的权限设置的一部分。

```
"permissions":
[
    "tabs",
    "bookmarks",
    "http://www.blogger.com/",
    "http://*.google.com/",
    "unlimitedStorage"
],
```

下面的表格列举了一个扩展或者app可以使用的权限。

注意：托管的app能使用权限:xxx,其他的都不能使用。

Permission	Description
match pattern	Specifies a*host permission*. Required if the extension wants to interact with the code running on pages. Many extension capabilities, such as cross-origin XMLHttpRequests,programmatically injected content scripts, and the cookies API require host permissions. For details on the syntax, see Match Patterns.
"background"	Makes Chrome start up early and and shut down late, so that apps and extensions can have a longer life. When any installed hosted app, packaged app, or extension has "background" permission, Chrome runs (invisibly) as soon as the user logs into their computer—before the user launches Chrome. The "background" permission also makes Chrome continue running (even after its last window is closed) until the user explicitly quits Chrome. **Note:**Disabled apps and extensions are treated as if they aren't installed. You typically use the "background" permission with a background page or (for hosted apps) a background window.
"bookmarks"	Required if the extension uses the chrome.bookmarks module.
"chrome://favicon/"	Required if the extension uses the "chrome://favicon/*url*" mechanism to display the favicon of a page. For example, to display the favicon of http://www.google.com/, you declare the "chrome://favicon/" permission and use HTML code like this: `<img src='chrome://favicon/http://www.google.com/' />`
"contextMenus"	Required if the extension uses the chrome.contextMenus module.
"cookies"	Required if the extension uses the chrome.cookies module.
"experimental"	Required if the extension uses any chrome.experimental.* APIs.
"geolocation"	Allows the extension to use the proposed HTML5 geolocation API without prompting the user for permission.
"history"	Required if the extension uses the chrome.history module.
"idle"	Required if the extension uses the chrome.idle module.
"management"	Required if the extension uses the chrome.management module.
"notifications"	Allows the extension to use the proposed HTML5 notification API without calling permission methods (such ascheckPermission()). For more information see Desktop Notifications.
"tabs"	Required if the extension uses the chrome.tabs or chrome.windows module.
"unlimitedStorage"	Provides an unlimited quota for storing HTML5 client-side data, such as databases and local storage files. Without this permission, the extension is limited to 5 MB of local storage. **Note:**This permission applies only to Web SQL Database and application cache (see issue 58985). Also, it doesn't currently work with wildcard subdomains such ashttp://*.example.com.

requirements

指定本app或扩展所需的特殊技术功能。安装扩展时，扩展商店根据这个清单，必要时劝阻用户在不支持所需功能的电脑上安装这些扩展。

目前只支持指定“3D”，也就是GPU加速。您可以指定所需的3D相关功能，比如：

```
"requirements": {
    "3D": {
      "features": ["css3d", "webgl"]
    }
  }
```

"css3d"的详细信息请参考CSS 3D Transforms 规范，"webgl"请参考WebGL API。 Chrome 3D功能的支持情况请参考WebGL and 3D graphics。 未来可能会增加更多技术功能的检测指定。

version

扩展的版本用一个到4个数字来表示，中间用点隔开。这些数字有些规则：必须在0到65535之间，非零数字不能0开头，比如，99999和032是不合法的。

下面是一些版本字符串例子：

- "version": "1"
- "version": "1.0"
- "version": "2.10.2"
- "version": "3.1.2.4567"

自动升级系统将比较版本来确定一个已经安装的扩展是否需要升级。如果一个发布的扩展有一个更新的版本字符串，比一个安装的扩展，这个扩展将自动升级。

版本字符串从比较从左边开始。如果这些数字相等，这个数字的右边的数字将被比较，这样持续下去。比如：1.2.0就比1.1.9.9999更新。

缺少的数字将用0来代替。例子，1.1.9.9999就比1.1.更新。

具体信息请参考Autoupdating。

manifest_version

用整数表示manifest文件自身格式的版本号。从Chrome 18开始，开发者应该（不是必须，但是2012年底左右就必须了）指定版本号为2（没有引号），如下所示：

```
"manifest_version": 2
```

manifest版本1从Chrome 18才开始逐步被弃用，版本2目前并不是必须的，但预计我们将在2012年底强制只支持版本2。还没有准备好支持manifest版本2的扩展、应用和主题，可以明确指定版本1，或者索性不提供本字段。

版本1到2之间的变化细节可以参考 `manifest_version` 文档。

在Chrome17（极速5.2）或之前的指定版本2将会发生不可预料的事情。

web_accessible_resources

一组字符串，指定本扩展在注入的目标页面上所需使用的资源的路径（相对于扩展的安装根目录）。例如，扩展在example.com上注入脚本以构建制界面，将其间所需的资源（图片、图标、样式、脚本等）加入白名单，如下所示：

```
{
    ...
    "web_accessible_resources": [
      "images/my-awesome-image1.png",
      "images/my-amazing-icon1.png",
      "style/double-rainbow.css",
      "script/double-rainbow.js"
    ],
    ...
  }
```

这些资源的访问URL是 `chrome-extension://[PACKAGE ID]/[PATH]` ，可通过调用 `chrome.extension.getURL` 构造出。 这些白名单资源是通过CORS头提供的，因此可被类似XHR这样的机制使用。

扩展的content scripts自身不需要加入白名单

资源的缺省可用性

manifest_version为2的扩展，缺省将不能使用除web_accessible_resources中指定外的任何其它任何扩展包内资源。

manifest*version为1*的扩展，缺省仍可访问任何扩展包内资源。但是_，一旦指定web_accessible_resources，将也只能访问其中指定的资源。

模式匹配

内容脚本可以作用到模式匹配定义好的URL集合上. 你能对manifest文件的内容脚本段的部分进行一个或多个模式匹配操作. 这里描述模式匹配语法 — 当你指定内容脚本将影响哪些URL时你需要遵循的规则.

任意一个模式匹配本质上都是一个以认可的协议(例如：http, https, file, ftp 或者 chrome-extension)开头的URL， 只是URL你可以包含"*"字符. 这里有一种特殊的模式匹配<all_urls>，它表示所有已认可的协议开头的URL都被匹配. 每个模式匹配都由以下三部分构成:</all_urls>

- 协议*(scheme)*— 例如,http or file or*

  注意:file协议不是缺省的(访问文件时可能需要明确指明使用file协议). 用户可以访问扩展管理页面或者设置页面来查看每个扩展独立的file协议设置.

- 域名*(host)*— 例如,www.google.com或者*.google.com*或者; 如果使用file协议, 这里就不需要域名部分

- 路径*(path)*— 例如,/,*/foo*, 或者/foo/bar

下边列出了基本语法:

```
_<url-pattern>_ := _<scheme>_://_<host>__<path>_
_<scheme>_ := '*' | 'http' | 'https' | 'file' | 'ftp' | 'chrome-extension'
_<host>_ := '*' | '*.' _<除 '/' 和 '*' 外的其它任意字符>_+
_<path>_ := '/' _<任意字符>_
```

''的含意依赖于它是出现在协议,域名, 或者路径中的那个部分*. 如果协议部分是*, 那么它表示匹配以*http*和*https*协议开头的*URL*. 如果域名部分是*, 那么它表示匹配任何域名*. 如果域名部分是*.域名, 那么它表示匹配该域名及任何该域名下的子域名. 而在路径部分， 每个'*'表示0个或多个字符. 下面的表展示了一些合法的模式匹配.

模式匹配	含意	能够匹配上的URL例子
http://*/*	匹配任何http协议的URL	http://www.google.com/ http://example.org/foo/bar.ht
http://*/foo*	匹配任何使用http协议的任何域名下，只要以/foo开头的URL	http://example.com/foo/bar.h http://www.google.com/foo
https://*.google.com/foo*bar	匹配任何使用https协议的google.com域名或其下子域名(例如www.google.com, docs.google.com, 或者google.com), 只要路径是以/foo开头，以bar结尾的URL	http://www.google.com/foo/b http://docs.google.com/foob
http://example.org/foo/bar.html	匹配指定的URL	http://example.org/foo/bar.ht
file:///foo*	匹配以/foo开头的任意本地文件	file:///foo/bar.html file:///foo
http://127.0.0.1/*	匹配任意以http协议的主机ip是127.0.0.1的URL	http://127.0.0.1/ http://127.0.0.1/foo/bar.html
://mail.google.com/	匹配任意以http://mail.google.com或者https://mail.google.com开头的URL.	http://mail.google.com/foo/b https://mail.google.com/foob
<all_urls></all_urls>	匹配所有认可的协议的URL. (参看这一段开头的认可协议列表，里边包含了所有认可的协议.)	http://example.org/foo/bar.ht file:///bar/baz.html

这里展示一些不合法的模式匹配的例子:

错误的模式匹配	错误原因
http://www.google.com	没有说明路径
http://*foo/bar	在域名部分使用'*'时，'*'后边只能是'.' 或者 '/'
http://foo.*.bar/baz	如果'*'出现在*host*部分时，'*'只能出现在开头
http:/bar	少写了协议分隔符("/" 应该是"//")
foo://*	无效的协议

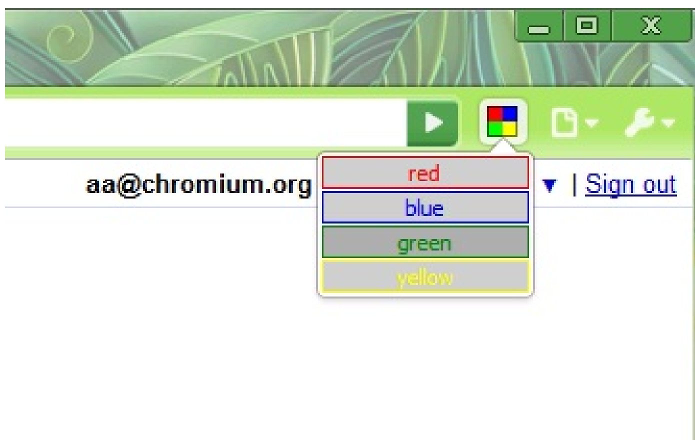
改变浏览器外观

Browser Actions

Contents

用 browser actions 可以在chrome主工具条的地址栏右侧增加一个图标。作为这个图标的延展，一个browser action图标还可以有tooltip、badge和popup。

如下图，地址栏右侧的彩色正方形是一个browser action的图标，图标下面的是popup。



如果你想创建一个不总是可见的图标，可以使用page action来代替browser action.

注意:Packaged apps 不能使用browser actions.

Manifest

在extension manifest中用下面的方式注册你的browser action:

```
{
  "name": "My extension",
  ...
  **"browser_action": {
    "default_icon": "images/icon19.png", _// optional_
    "default_title": "Google Mail",      _// optional; shown in tooltip_
    "default_popup": "popup.html"        _// optional_
  }**,
  ...
}
```

UI的组成部分

一个 browser action 可以拥有一个图标，一个tooltip，一个badge和一个popup。

图标

Browser action 图标推荐使用宽高都为19像素，更大的图标会被缩小。

你可以用两种方式来设置图标: 使用一个静态图片或者使用HTML5canvas element。 使用静态图片适用于简单的应用程序，你也可以创建诸如平滑的动画之类更丰富的动态UI(如canvas element)。

静态图片可以是任意WebKit支持的格式，包括 BMP，GIF，ICO，JPEG或 PNG。

修改**browser_action**的manifest中 **default_icon**字段，或者调用setIcon()方法。

ToolTip

修改**browser_action**的manifest中**default_title**字段，或者调用setTitle()方法。你可以为**default_title**字段指定本地化的字符串；点击Internationalization查看详情。

Badge

Browser actions可以选择性的显示一个*badge*— 在图标上显示一些文本。Badges 可以很简单的为browser action更新一些小的扩展状态提示信息。

因为badge空间有限，所以只支持4个以下的字符。

设置badge文字和颜色可以分别使用setBadgeText()andsetBadgeBackgroundColor()。

Popup

如果browser action拥有一个popup，popup 会在用户点击图标后出现。popup 可以包含任意你想要的HTML内容，并且会自适应大小。

在你的browser action中添加一个popup，创建弹出的内容的HTML文件。 修改**browser_action**的manifest中**default_popup**字段来指定HTML文件， 或者调用setPopup()方法。

Tips

为了获得最佳的显示效果, 请遵循以下原则：

- 确认 Browser actions 只使用在大多数网站都有功能需求的场景下。
- 确认 Browser actions 没有使用在少数网页才有功能的场景， 此场景请使用page actions。
- 确认你的图标尺寸尽量占满19x19的像素空间。 Browser action 的图标应该看起来比page action的图标更大更重。
- 不要尝试模仿Google Chrome的扳手图标，在不同的themes下它们的表现可能出现问题,， 并且扩展应该醒目些。
- 尽量使用alpha通道并且柔滑你的图标边缘，因为很多用户使用themes，你的图标应该在在各种背景下都表现不错。
- 不要不停的闪动你的图标，这很惹人反感。

范例

你可以在 examples/api/browserAction目录找到简单的browser actions范例，更多的范例和帮助文档可以参考Samples。

API reference: chrome.browserAction

Properties

getLastError

chrome.extensionlastError

Methods

setBadgeBackgroundColor

void chrome.browserAction.setBadgeBackgroundColor(, object `details`)

Undocumented.

设置badge的背景颜色。

Parameters

`details` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`color` *(optional enumerated Type array of Type array of integer paramType)*

Undocumented.

范围为[0,255]整数构成的结构，用来描述badge的RGBA颜色。例如：不透明的红色是[255, 0, 0, 255]。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`tabId` *(optional enumerated Type array of integer)*

Undocumented.

可选参数，将设置限制在被选中的标签，当标签关闭时重置。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

Callback function

The callback *parameter* should specify a function that looks like this:

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(Type param1, Type param2) {...};
```

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

setBadgeText

void chrome.browserAction.setBadgeText(, object `details`)

Undocumented.

设置browser action的badge文字，badge 显示在图标上面。

Parameters

`details` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`text` *(optional enumerated Type array of string)*

Undocumented.

任意长度的字符串，但只有前4个字符会被显示出来。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`tabId` *(optional enumerated Type array of integer)*

Undocumented.

可选参数，将设置限制在被选中的标签，当标签关闭时重置。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser

version.

Returns

Callback function

The callback *parameter* should specify a function that looks like this:

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(Type param1, Type param2) {...};
```

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

setIcon

void chrome.browserAction.setIcon(, object `details`)

Undocumented.

设置browser action的图标。图标可以是一个图片的路径或者是从一个canvas元素提取的像素信息.。无论是图标路径还是canvas的 **imageData**，这个属性必须被指定。

Parameters

`details` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`imageData` *(optional enumerated Type array of ImageData)*

Undocumented.

图片的像素信息。必须是一个ImageData 对象(例如：一个canvas元素)。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`path` *(optional enumerated Type array of string)*

Undocumented.

图片在扩展中的的相对路径。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`tabId` *(optional enumerated Type array of integer)*

Undocumented.

可选参数，将设置限制在被选中的标签，当标签关闭时重置。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

Callback function

The callback *parameter* should specify a function that looks like this:

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(Type param1, Type param2) {...};
```

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

setPopup

void chrome.browserAction.setPopup(, object `details`)

Undocumented.

设置一个点击browser actions时显示在popup中的HTML。

Parameters

`details` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`tabId` *(optional enumerated Type array of integer)*

Undocumented.

可选参数，将设置限制在被选中的标签，当标签关闭时重置。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`popup` *(optional enumerated Type array of string)*

Undocumented.

popup中显示的html文件。如果设置为空字符(''), 将不显示popup。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

Callback function

The callback *parameter* should specify a function that looks like this:

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(Type param1, Type param2) {...};
```

这个功能已经在**chromium 5.0.316.0**版本添加。如果你需要这个功能，可以通过manifest的minimum_chrome_version键值来确认你的扩展不会运行在早期的浏览器版本。

setTitle

void chrome.browserAction.setTitle(, object `details`)

Undocumented.

设置browser action的标题，这个将显示在tooltip中。

Parameters

`details` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`title` *(optional enumerated Type array of string)*

Undocumented.

鼠标移动到browser action上时显示的文字。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`tabId` *(optional enumerated Type array of integer)*

Undocumented.

可选参数，将设置限制在被选中的标签，当标签关闭时重置。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

Callback function

The callback *parameter* should specify a function that looks like this:

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(Type param1, Type param2) {...};
```

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Events

onClicked

chrome.browserAction.onClicked.addListener(function(Tab tab) {...});

Undocumented.

当browser action 图标被点击的时候触发，当browser action是一个popup的时候，这个事件将不会被触发。

Parameters

`tab` *(optional enumerated Tab array of paramType)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Types

type name

Context Menus

内容

Context菜单用于在Chrome的右键菜单中增加自己的菜单项。

您可以选择针对不同类型的对象（如图片，链接，页面）增加右键菜单项。

您可以根据需要添加多个右键菜单项。一个扩展里添加的多个右键菜单项会被Chrome自动组合放到对应扩展名称的二级菜单里。

右键菜单可以出现在任意文档（或文档中的框架）中，甚至是本地文件（如file://或者Chrome://）中。若想控制右键菜单在不同文档中的显示，可以在调用create()和update()时指定documentUrlPatterns。

版本说明： 低于Chrome 14的版本,右键菜单只能用于http:// 或者 https:// 类型的文档。

清单

要使用contentMenus API，您必须在清单中声明“contentMenus”权限。同时，您应该指定一个16x16的图标用作右键菜单的标识。例如：

```
{
        "name": "My extension",
        ...
        "permissions": [
          **"contextMenus"**
        ],
        "icons": {
          **"16": "icon-bitty.png",**
          "48": "icon-small.png",
          "128": "icon-large.png"
        },
        ...
}
```


范例

您可以在代码例子页面找到使用contentMenus API的简单范例。

API 参考: Chrome.contextMenus

方法

create

integer Chrome.contextMenus.create(object `createProperties` , function `callback`)

创建一个新的右键菜单项。注意：如果在创建的过程中出现错误，会在回调函数触发后才能捕获到，错误详细信息保存在Chrome.extension.lastError中。

参数

`createProperties` *(object)*

Undocumented.

`type` *(optional enumerated string ["normal", "checkbox", "radio", "separator"])*

右键菜单项的类型。默认为“normal”。

`title` *(optional string)*

右键菜单项的显示文字；除非为“separator”类型，否则此参数是必须的。如果类型为“selection”，您可以在字符串中使用 `%s` 显示选定的文本。例如，如果参数的值为"Translate '%s' to Pig Latin"，而用户选中了文本“cool”，那么显示在菜单中的将会是"Translate 'cool' to Pig Latin"。

`checked` *(optional boolean)*

Checkbox或者radio的初始状态：true代表选中，false代表未选中。在给定的radio中只能有一个处于选中状态。

`contexts` *(optional array of string ["all", "page", "frame", "selection", "link", "editable", "image", "video", "audio"])*

右键菜单项将会在这个列表指定的上下文类型中显示。默认为“page”。

`onclick` *(optional function)*

当菜单项被点击时触发的函数。

参数

`info` *(OnClickData)*

右键菜单项被点击时相关的上下文信息。

`tab` *(Tab)*

右键菜单项被点击时，当前标签的详细信息。

`parentId` *(optional integer)*

右键菜单项的父菜单项ID。指定父菜单项将会使此菜单项成为父菜单项的子菜单。

`documentUrlPatterns` *(optional array of string)*

这使得右键菜单只在匹配此模式的url页面上生效（这个对框架也适用）。详细的匹配格式见：模式匹配页面。

`targetUrlPatterns` *(optional array of string)*

类似于documentUrlPatterns，但是您可以针对img/audio/video标签的src属性和anchor标签的href做过滤。

`enabled` *(optional boolean)*

启用或者禁用此菜单项，启用为true，禁用为false。默认为true。

`callback` *(optional function)*

在创建完菜单项后触发。如果创建过程中有错误产生，其详细信息在Chrome.extension.lastError中。

返回

(integer)

新创建右键菜单项的ID。

回调

如果需要指定回调函数，则回调函数格式如下：

```
function() {...};
```

remove

Chrome.contextMenus.remove(integer `menuItemId`, function `callback`)

删除一个右键菜单。

参数

`menuItemId` *(integer)*

待删除的右键菜单项的ID

`callback` *(optional function)*

当右键菜单项被删除后触发。

回调

如果需要指定回调函数，则回调函数格式如下：

```
function() {...};
```

removeAll

Chrome.contextMenus.removeAll(function `callback`)

删除此扩展添加的所有右键菜单项。

参数

`callback` *(optional function)*

删除完成后触发。

回调

如果需要指定回调函数，则回调函数格式如下：

```
function() {...};
```

update

Chrome.contextMenus.update(integer `id` , object `updateProperties` , function `callback`)

更新已创建的右键菜单项。

参数

`id` *(integer)*

待更新的右键菜单项的ID.

`updateProperties` *(object)*

待更新的属性。与创建右键菜单项时的属性参数一样。

`type` *(optional enumerated string ["normal", "checkbox", "radio", "separator"])*

Undocumented.

`title` *(optional string)*

Undocumented.

`checked` *(optional boolean)*

Undocumented.

`contexts` *(optional array of string ["all", "page", "frame", "selection", "link", "editable", "image", "video", "audio"])*

Undocumented.

`onclick` *(optional function)*

Undocumented.

`parentId` *(optional integer)*

注意：不能将右键菜单项设置成自己子菜单的子菜单。

`documentUrlPatterns` *(optional array of string)*

Undocumented.

`targetUrlPatterns` *(optional array of string)*

Undocumented.

`enabled` *(optional boolean)*

Undocumented.

`callback` *(optional function)*

右键菜单项更新完成后触发。

回调

如果需要指定回调函数，则回调函数格式如下：

```
function() {...};
```

类型

OnClickData

(object)

当右键菜单项被点击时的信息。

`menuItemId` *(integer)*

被点击的右键菜单项的ID。

`parentMenuItemId` *(optional integer)*

被点击的右键菜单项的父菜单（如果存在）ID。

`mediaType` *(optional string)*

点击激活此右键菜单项时，被点击的元素的类型，如:'image', 'video'或者 'audio'。

`linkUrl` *(optional string)*

链接的url（如果被点击的元素是链接）。

`srcUrl` *(optional string)*

如果被点击元素有 'src' 属性。

`pageUrl` *(string)*

点击所在页面的URL。

`frameUrl` *(optional string)*

框架元素的URL（如果点击的元素是一个框架）。

`selectionText` *(optional string)*

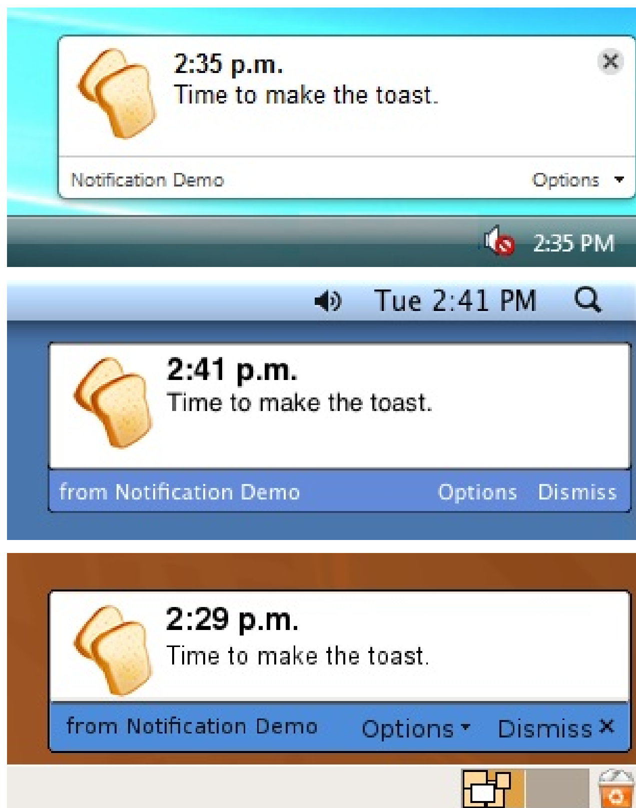
如果点击时选择了文本，则为选中的文本内容。

`editable` *(string)*

被点击的元素是否可编辑，比如文本输入框就是可编辑的。

桌面通知

通知用户发生了一些重要的事情。桌面通知会显示在浏览器窗口之外。 下面的图片是通知显示时的效果，在不同的平台下，通知的显示效果会有些细微区别。



通常直接使用一小段 JavaScript 代码创建通知，当然也可以通过扩展包内的一个单独HTML页面。

声明

可以在 extension manifest 中声明使用通知权限，像这样：

```
{
  "name": "My extension",
  ...
 **"permissions": [
    "notifications"
  ]**,
  ...
}
```

注意： 扩展声明的 `notifications` 权限总是允许创建通知。 这样申明之后就不再需要调用 `webkitNotifications.checkPermission()` 。

与扩展页面交互

扩展可以使用 getBackgroundPage() 和 getViews()在通知与扩展页面中建立交互。 例如：

```
// 在通知中调用扩展页面方法...
chrome.extension.getBackgroundPage().doThing();

// 从扩展页面调用通知的方法...
chrome.extension.getViews({type:"notification"}).forEach(function(win) {
  win.doOtherThing();
});
```

例子

一个简单的使用通知的例子，参见 examples/api/notifications 目录。 更多的例子，以及在查看代码中遇到的一些问题，请参见 代码例子。

也可以参考 html5rocks.com 的 通知指南。 如果你只是声明 "通知" 的权限，可以忽略权限相关的代码，它不是必要的。

API

扩展的桌面通知 API， 也可用于显示一个网页。 如以下代码所示，首先创建一个简单的文字通知或 HTML 通知，然后显示通知。

```
// 创建一个简单的文字通知：
var notification = webkitNotifications.createNotification(
  '48.png',  // icon url - can be relative
  'Hello!',  // notification title
  'Lorem ipsum...'  // notification body text
);

// 或者创建一个 HTML 通知：
var notification = webkitNotifications.createHTMLNotification(
  'notification.html'  // html url - can be relative
);

// 显示通知
notification.show();
```

完整的 API 详情，请参看 Desktop notifications draft specification。

API reference: chrome.apiname

Properties

getLastError

chrome.extensionlastError

Methods

method name

void chrome.module.methodName(, ``)

Undocumented.

A description from the json schema def of the function goes here.

Parameters

Returns

Callback function

The callback *parameter* should specify a function that looks like this:

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(Type param1, Type param2) {...};
```

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Events

event name

chrome.bookmarksonEvent.addListener(function(Type param1, Type param2) {...});

Undocumented.

A description from the json schema def of the event goes here.

Parameters

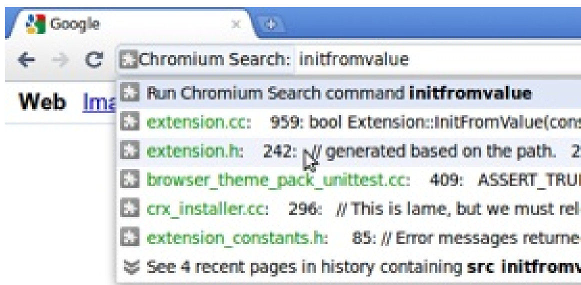
Types

type name

Omnibox

Contents

omnibox 应用程序界面允许向Google Chrome的地址栏注册一个关键字，地址栏也叫omnibox。



当用户输入你的扩展关键字，用户开始与你的扩展交互。每个击键都会发送给你的扩展，扩展提供建议作为相应的响应。

建议可以被格式化多种方式。当用户接受建议，你的扩展被通知可以执行动作。

Manifest

使用omnibox 应用程序界面，必须在 manifest 中包含omnibox 关键字段。需要指定像素为16x16的图标，以便当用户输入关键字时，在地址栏中显示。

如：

```
{
  "name": "Aaron's omnibox extension",
  "version": "1.0",
  **"omnibox": { "keyword" : "aaron" },**
  **"icons": {**
    **"16": "16-full-color.png"**
  **},**
  "background_page": "background.html"
}
```

提示: Chrome 自动创建灰度模式16x16像素的图标。你应该提供全色版本图标以便可以在其他场景下使用。 如：Context menus API

使用全色的16x16像素图标。

示例

从sample page 页面可以找到使用该API的例子。.

应用程序界面参考: chrome.omnibox

Methods

method name

void chrome.module.methodName(, ``)

Undocumented.

A description from the json schema def of the function goes here.

Parameters

Returns

Callback function

The callback *parameter* should specify a function that looks like this:

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(Type param1, Type param2) {...};
```

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

setDefaultSuggestion

void chrome.omnibox.setDefaultSuggestion(, object `suggestion`)

Undocumented.

设置缺省建议的描述和风格。缺省建议是显示在 URL地址栏下的第一个建议显示文字

Parameters

`suggestion` *(optional enumerated Type array of object)*

Undocumented.

一个局部的SuggestResult 对象，没有'content' 参数。关于该参数的描述，请参见SuggestResult。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`description` *(optional enumerated Type array of string)*

Undocumented.

显示在缺省建议中的文本，可以包含'%s'并可以被用户输入替换。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

Callback function

The callback *parameter* should specify a function that looks like this:

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(Type param1, Type param2) {...};
```

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Events

onInputCancelled

chrome.omnibox.onInputCancelled.addListener(function() {...});

Undocumented.

用户结束键盘输入会话，但未接受该输入（取消了输入）。

Parameters

onInputChanged

chrome.omnibox.onInputChanged.addListener(function(string text, function suggest) {...});

Undocumented.

用户修改了在 omnibox中的输入。

Parameters

`text` *(optional enumerated Type array of string)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`suggest` *(optional enumerated Type array of function)*

Undocumented.

一个传给onInputChanged 事件的回调，用来在事件发生的时候，发送回建议给浏览器。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Parameters

`paramName` *(optional enumerated Type array of SuggestResult array of paramType paramType)*

Undocumented.

建议结果，数组。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

onInputEntered

chrome.omnibox.onInputEntered.addListener(function(string text) {...});

Undocumented.

用户接收了omnibox中的数据。

Parameters

`text` *(optional enumerated Type array of string)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

onInputStarted

chrome.omnibox.onInputStarted.addListener(function() {...});

Undocumented.

用户输入扩展的关键字，开始了一个键盘输入会话。 这个事件在会话开始时发送，早于其它事件，而且一个会话只会发送一次。

Parameters

Types

SuggestResult

`paramName` *(optional enumerated Type array of object)*

Undocumented.

建议结果。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`content` *(optional enumerated Type array of string)*

Undocumented.

在URL区域中的文本，当用户选择该条目时发送给扩展。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`description` *(optional enumerated Type array of string)*

Undocumented.

The URL下拉列表中显示的文本。可以包含一个XML风格标记。支持的标签是'url' (作为一个文法上的URL)， 'match' (作为匹配用户请求数据的高亮文本显示)，以及 'dim' (作为灰色辅助文本)。风格可以嵌套。

选项页

为了让用户设定你的扩展功能，你可能需要提供一个选项页。如果你提供了选项页，在扩展管理页面 chrome://extensions上会提供一个链接。点击选项链接就可以打开你的选项页。

在manifest中定义你的选项页

```
{
  "name": "My extension",
  ...
  **"options_page": "options.html"**,
  ...
}
```

编写你的选项页

下面是个选项页的范例:

```
<html>
<head><title>My Test Extension Options</title></head>
<script type="text/javascript">
// Saves options to localStorage.
function save_options() {
  var select = document.getElementById("color");
  var color = select.children[select.selectedIndex].value;
  localStorage["favorite_color"] = color;
  // Update status to let user know options were saved.
  var status = document.getElementById("status");
  status.innerHTML = "Options Saved.";
  setTimeout(function() {
    status.innerHTML = "";
  }, 750);
}
// Restores select box state to saved value from localStorage.
function restore_options() {
  var favorite = localStorage["favorite_color"];
  if (!favorite) {
    return;
  }
  var select = document.getElementById("color");
  for (var i = 0; i < select.children.length; i++) {
    var child = select.children[i];
    if (child.value == favorite) {
      child.selected = "true";
      break;
    }
  }
}
</script>
<body onload="restore_options()">
Favorite Color:
<select id="color">
 <option value="red">red</option>
 <option value="green">green</option>
 <option value="blue">blue</option>
 <option value="yellow">yellow</option>
</select>
<br>
<button onclick="save_options()">Save</button>
</body>
</html>
```

注意事项

- 本功能在Chromium4.0.222.x以上版本生效。
- 为了鼓励不同扩展之间的选项页的一致性，我们计划提供一些默认的css样式。你可以关注 crbug.com/25317的更新。

Override替代页

使用替代页，可以将Chrome默认的一些特定页面替换掉，改为使用扩展提供的页面。这让扩展开发者可以开发更多有趣或者实用的基本功能页面。替代页通常会有大量的CSS和JavaScript代码。

扩展可以替代如下页面：

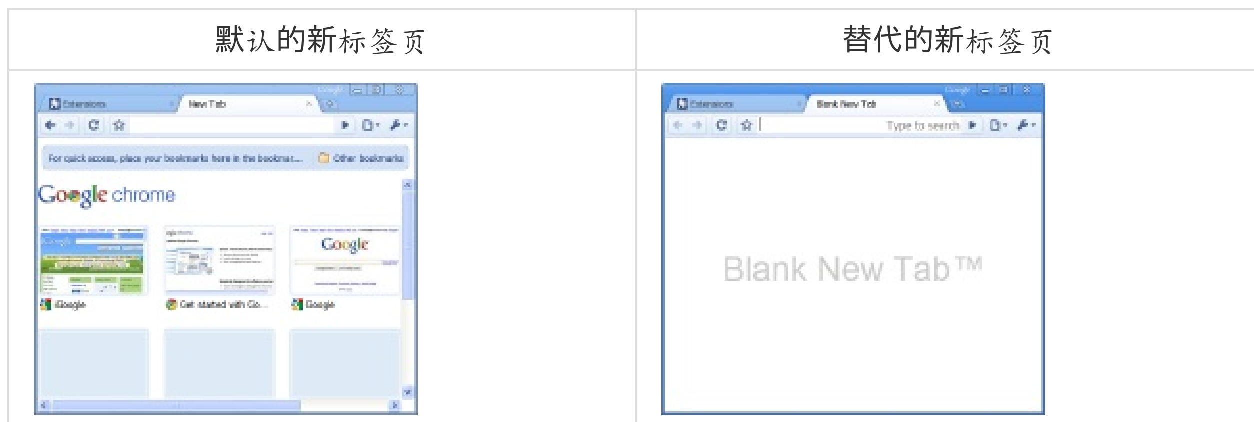
- 书签管理器：从工具菜单上点击书签管理器时访问的页面，或者从地址栏直接输入 **chrome://bookmarks**。
- 历史记录：从工具菜单上点击历史记录时访问的页面，或者从地址栏直接输入 **chrome://history**。
- 新标签页：当创建新标签的时候访问的页面，或者从地址栏直接输入 **chrome://newtab**。

注意：一个扩展只能替代一个页面。

注意：如果你替代隐身窗口的页面，请注意要在manifest中将 incognito 属性指定为 "spanning"。

注意：你不能替代隐身窗口的新标签页。

下面的截图是默认的新标签页和被扩展替换掉的新标签页。

默认的新标签页	替代的新标签页
	

Manifest

下面是在extension manifest中注册替代页的写法。

```
{
  "name": "My extension",
  ...
 **"chrome_url_overrides" : {
    "_pageToOverride_": "_myPage.html_"
  }**,
  ...
}
```

对于示例中的*pageToOverride*，可替换成如下关键字

- bookmarks
- history
- newtab

提示

要制作一个优秀的替代页，请一定遵循如下指导原则：

- 你的页面实现的要即小又快。 用户希望内建的浏览器页面可以快速的打开，请避免做一些要消耗很多时间的事。如：尽量避免从网络中或者数据库资源中提取数据。

- 页面要带标题。 否则用户可能会看到网页的URL，造成困扰。其实就是在页面头上加入：<title>NewTab</title>

- 别指望页面会获得键盘输入焦点。 通常新标签创建的时候，地址栏会获得输入焦点，而不是页面。

- 别试着模仿默认的新标签页。 对于新标签页上的重要功能支持，如：最近关闭的标签、主题背景图等，APIs的支持尚未完善，在这些APIs确认完备之前，你最好做一些完全不同的新标签页设计（以避免使用这些非正式的APIs）。

范例

你可以在这里找到范例examples/api/override，其他范例在这里Samples

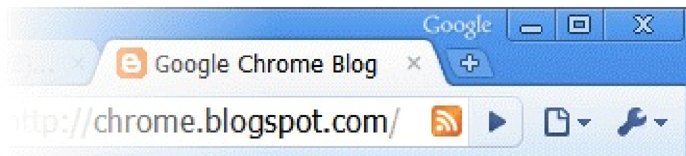
Page Actions

Contents

使用page actions把图标放置到地址栏里。page actions定义需要处理的页面的事件，但是它们不是适用于所有页面的。下面是一些page actions的示例：

- 订阅该页面的RSS feed
- 为页面的图片做一个幻灯片

在下面的屏幕截图中的RSS图标，提供了一个可以让用户订阅当前页面RSS Feed的page action。



想让扩展图标总是可见，则使用browser action。

注意： 打包的应用程序不能使用page actions。

Manifest

在 extension manifest 中用下面的方式注册你的page action：

```
{
  "name": "My extension",
  ...
  **"page_action": {
    "default_icon": "icons/foo.png", _// optional_
    "default_title": "Do action",    _// optional; shown in tooltip_
    "default_popup": "popup.html"    _// optional_
  }**,
  ...
}
```

UI的组成部分

同browser actions一样，page actions 可以有图标、提示信息、 弹出窗口。但没有badge，也因此，作为辅助，page actions可以有显示和消失两种状态。阅读browser action UI. 可以找到图标、提示信息、 弹出窗口相关信息。

使用方法 show() 和 hide() 可以显示和隐藏page action。缺省情况下page action是隐藏的。当要显示时，需要指定图标所在的标签页，图标显示后会一直可见，直到该标签页关闭或开始显示不同的URL （如：用户点击了一个连接）

提示

为了获得最佳的视觉效果，请遵循下列指导方针：

- 要只对少数页面使用page action；
- 不要对大多数页面使用它，如果功能需要，使用 browser actions代替。
- 没事别总让图标出现动画，那会让人很烦。

示例

你可以在examples/api/pageAction 找到使用page action的简单示例，其它例子和源代码帮助查看Samples。

API reference: chrome.pageAction

Properties

getLastError

chrome.extensionlastError

Methods

hide

void chrome.pageAction.hide(, integer `tabId`)

Undocumented.

隐藏page action.

Parameters

`tabId` *(optional enumerated Type array of integer)*

Undocumented.

要执行这个动作的标签ID。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

Callback function

The callback *parameter* should specify a function that looks like this:

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(Type param1, Type param2) {...};
```

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

setIcon

void chrome.pageAction.setIcon(, object `details`)

Undocumented.

为page aciton设置图标。图标可以是一个图片的路径或者是从一个canvas元素提取的像素信息.。无论是图标路径还是canvas的 **imageData**，这个属性必须被指定。

Parameters

`details` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`tabId` *(optional enumerated Type array of integer)*

Undocumented.

要执行这个动作的标签ID。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`imageData` *(optional enumerated Type array of ImageData)*

Undocumented.

图片的像素信息。必须是一个ImageData 对象(例如：一个canvas元素)。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`path` *(optional enumerated Type array of string)*

Undocumented.

图片在扩展中的的相对路径。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`iconIndex` *(optional enumerated Type array of integer)*

Undocumented.

不建议。manifest中定义的，0开始的图标数组下标。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

Callback function

The callback *parameter* should specify a function that looks like this:

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(Type param1, Type param2) {...};
```

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

setPopup

void chrome.pageAction.setPopup(, object `details`)

Undocumented.

设置一个点击page actions时显示在popup中的HTML。

Parameters

`details` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`tabId` *(optional enumerated Type array of integer)*

Undocumented.

要设置popup的标签ID。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`popup` *(optional enumerated Type array of string)*

Undocumented.

popup中显示的html文件。如果设置为空字符(''), 将不显示popup。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

Callback function

The callback *parameter* should specify a function that looks like this:

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(Type param1, Type param2) {...};
```

这个功能已经在**chromium 5.0.316.0**版本添加。如果你需要这个功能，可以通过manifest的minimum_chrome_version键值来确认你的扩展不会运行在早期的浏览器版本。

setTitle

void chrome.pageAction.setTitle(, object `details`)

Undocumented.

设置page action的标题，这将显示在tooltip中。

Parameters

`details` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser

version.

`tabId` *(optional enumerated Type array of integer)*

Undocumented.

要设置标题的标签ID。

`title` *(optional enumerated Type array of string)*

Undocumented.

提示信息字符串。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

Callback function

The callback *parameter* should specify a function that looks like this:

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(Type param1, Type param2) {...};
```

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

show

void chrome.pageAction.show(, integer `tabId`)

Undocumented.

显示page action，无论标签是否被选中。

Parameters

`tabId` *(optional enumerated Type array of integer)*

Undocumented.

要执行这个动作的标签ID。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

Callback function

The callback *parameter* should specify a function that looks like this:

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(Type param1, Type param2) {...};
```

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Events

onClicked

chrome.pageAction.onClicked.addListener(function(Tab tab) {...});

Undocumented.

当page action图标被点击的时候调用，如果page action是一个popup，这个事件将不会触发。Fi

Parameters

`tab` *(optional enumerated Tab array of paramType)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Types

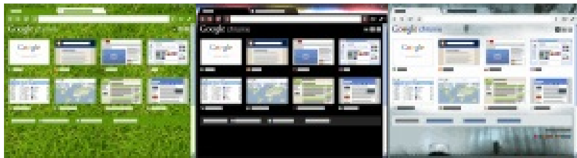
type name

主题

Contents

主题是一种特殊的扩展,可以用来改变整个浏览器的外观。主题和标准扩展的打包方式类似,但是主题中不能包含JavaScript或者HTML代码。

您可以更换主题在主题画廊。



Manifest

下面是manifest.json的示例代码，用来生成一个特定的主题。

```
{
  "version": "2.6",
  "name": "camo theme",
**  "theme": {
    "[images](#images)" : {
      "theme_frame" : "images/theme_frame_camo.png",
      "theme_frame_overlay" : "images/theme_frame_stripe.png",
      "theme_toolbar" : "images/theme_toolbar_camo.png",
      "theme_ntp_background" : "images/theme_ntp_background_norepeat.png",
      "theme_ntp_attribution" : "images/attribution.png"
    },
    "[colors](#colors)" : {
      "frame" : [71, 105, 91],
      "toolbar" : [207, 221, 192],
      "ntp_text" : [20, 40, 0],
      "ntp_link" : [36, 70, 0],
      "ntp_section" : [207, 221, 192],
      "button_background" : [255, 255, 255]
    },
    "[tints](#tints)" : {
      "buttons" : [0.33, 0.5, 0.47]
    },
    "[properties](#properties)" : {
      "ntp_background_alignment" : "bottom"
    }
  }**
}
```

颜色

颜色采用 RGB格式。 如果想了解更多颜色值，可以在 theme_service.cc 文件中查找 kDefaultColor* 。

图片

图片资源使用扩展的根目录作为引用路径。你可以通过 theme_service.cc .文件中的 kThemeableImages 修改任何图片资源。删除 "IDR_" 标识并将剩下的字符串修改为小写字母。例如， 将 IDR_THEME_NTP_BACKGROUND (表示kThemeableImages用来指定新标签的背景) 修改为"theme_ntp_background"。

属性

该字段允许您指定主题的属性。例如背景对齐、背景重复和logo的轮换。更多的信息可以参考 theme_service.cc 。

tints

您可以将tints应用到按钮、框架UI 、背景标签等用户界面。Google Chrome 支持tints，但是不支持图片。因为图片不能跨平台工作，并且当添加一些新的按钮时会变得很脆弱。 如果想了解tints，在 theme_service.cc 文件 "tints" 字段中查找 kDefaultTint*。

Tints 采用Hue-Saturation-Lightness (HSL)格式, 浮点数范围为0 - 1.0：

- **Hue** 是一个绝对值，只包含0 和1 。
- **Saturation** 是相对于当前图片的。0.5 表示不变，0表示完全稀释，1表示全部饱和。
- **Lightness** 也是一个相对值。0.5 表示不变，0表示有像素是黑色，1表示所有像素是白色。

你可以选择-1。0表示任何 HSL的值都不变。

附加文档

社区提供了一些文档用来帮助开发主题。地址如下：

> http://code.google.com/p/chromium/wiki/ThemeCreationGuide

与浏览器交互

书签

内容

使用chrome.bookmarks模块来创建、组织和管理书签。也可参看 Override Pages，来创建一个可定制的书签管理器页面。



介绍

您必须在扩展说明文件中声明使用书签API的权限。例如：

```
{
  "name": "My extension",
  ...
  **"permissions": [
    "bookmarks"
  ]**,
  ...
}
```

对象和属性

书签是按照树状结构组织的，每个节点都是一个书签或者一组节点（每个书签夹可包含多个节点）。每个节点都对应一个 BookmarkTreeNode 对象。

可以通过 chrome.bookmarks API来使用BookmarkTreeNode的属性。例如，当调用函数 create()，可以传入参数新节点的父节点（父节点ID），以及可选的节点索引，标题和url属性。可参看 BookmarkTreeNode 来获取节点的信息。

例子

下面的 代码创建了一个标题为 "Extension bookmarks"的书签夹。函数create()的第一个参数指定了新书签夹的属性，第二个参数定义了一个在书签夹创建后要执行的回调函数

```
chrome.bookmarks.create({'parentId': bookmarkBar.id,
                         'title': 'Extension bookmarks'},
                        function(newFolder) {
  console.log("added folder: " + newFolder.title);
});
```

接下来的代码创建了一个指向扩展开发文档的书签。如果创建书签失败，也不会引起什么问题，所以没有指定回调函数。

```
chrome.bookmarks.create({'parentId': extensionsFolderId,
                         'title': 'Extensions doc',
                         'url': 'http://code.google.com/chrome/extensions'});
```

使用该API的实例请参看 basic bookmarks sample。其他例子和源码请参看 Samples。

API 参考：chrome.bookmarks

Properties

getLastError

chrome.extensionlastError

方法

create

void chrome.bookmarks.create(, object `bookmark` , function `callback`)

Undocumented.

在指定父节点下创建一个书签或者书签夹。如果url为空，则创建一个书签夹。

参数

`bookmark` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`parentId` 父节点ID*(optional enumerated Type array of string)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`index` *(optional enumerated Type array of integer)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`title` *(optional enumerated Type array of string)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`url` *(optional enumerated Type array of string)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`callback` *(optional enumerated Type array of function)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

回调函数

The callback *parameter* should specify a function that looks like this:

如果需要指定回调函数，则回调函数格式如下：

```
function(BookmarkTreeNode result) {...};
```

`result` *(optional enumerated BookmarkTreeNode array of paramType)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

method name

void chrome.module.methodName(, ``)

Undocumented.

A description from the json schema def of the function goes here.

Parameters

Returns

Callback function

The callback *parameter* should specify a function that looks like this:

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(Type param1, Type param2) {...};
```

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

get

void chrome.bookmarks.get(, string or array of string `idOrIdList` , function `callback`)

Undocumented.

获取指定的书签节点。

参数

`idOrIdList` *(optional enumerated Type array of string or array of string)*

Undocumented.

一个字符串类型的Id，或者字符串数组

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`callback` *(optional enumerated Type array of function)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

回调函数

如果需要指定回调函数，则回调函数格式如下：

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(array of BookmarkTreeNode results) {...};
```

`results` *(optional enumerated Type array of BookmarkTreeNode array of paramType paramType)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

getChildren

void chrome.bookmarks.getChildren(, string `id` , function `callback`)

Undocumented.

获取指定的书签节点的子节点

参数

`id` *(optional enumerated Type array of string)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`callback` *(optional enumerated Type array of function)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

回调函数

如果需要指定回调函数，则回调函数格式如下：

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(array of BookmarkTreeNode results) {...};
```

`results` *(optional enumerated Type array of BookmarkTreeNode array of paramType paramType)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

getRecent

void chrome.bookmarks.getRecent(, integer `numberOfItems` , function `callback`)

Undocumented.

获取最近添加的书签。

Parameters

`numberOfItems` *(optional enumerated Type array of integer)*

Undocumented.

最多返回的条目数量。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`callback` *(optional enumerated Type array of function)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

回调函数

如果需要指定回调函数，则回调函数格式如下：

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(array of BookmarkTreeNode results) {...};
```

`results` *(optional enumerated Type array of BookmarkTreeNode array of paramType paramType)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

getTree

void chrome.bookmarks.getTree(, function `callback`)

Undocumented.

按照层次结构获取所有书签。

Parameters

`callback` *(optional enumerated Type array of function)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

回调函数

回调参数 *parameter* 指定的回调函数如下：

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(array of BookmarkTreeNode results) {...};
```

`results` *(optional enumerated Type array of BookmarkTreeNode array of paramType paramType)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

method name

void chrome.module.methodName(, ``)

Undocumented.

A description from the json schema def of the function goes here.

Parameters

Returns

Callback function

The callback *parameter* should specify a function that looks like this:

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(Type param1, Type param2) {...};
```

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

move

void chrome.bookmarks.move(, string `id` , object `destination` , function `callback`)

Undocumented.

移动指定的书签节点到指定的位置。

Parameters

`id` *(optional enumerated Type array of string)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`destination` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`parentId` *(optional enumerated Type array of string)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`index` *(optional enumerated Type array of integer)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`callback` *(optional enumerated Type array of function)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

回调函数

如果需要指定*callback*参数，函数格式如下：

```
function(BookmarkTreeNode result) {...};
```

`result` *(optional enumerated BookmarkTreeNode array of paramType)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

remove

void chrome.bookmarks.remove(, string `id` , function `callback`)

Undocumented.

删除一个书签或者空书签夹。

Parameters

`id` *(optional enumerated Type array of string)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser

version.

`callback` *(optional enumerated Type array of function)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

回调函数

如果需要指定*callback*参数，函数格式如下：

```
function() {...};
```

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

removeTree

void chrome.bookmarks.removeTree(, string `id` , function `callback`)

Undocumented.

删除书签夹目录（和它的子目录）。

参数

`id` *(optional enumerated Type array of string)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`callback` *(optional enumerated Type array of function)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

回调函数

如果需要指定*callback*参数，函数格式如下：

```
function() {...};
```

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

search

void chrome.bookmarks.search(, string `query` , function `callback`)

Undocumented.

根据指定查询条件搜索书签节点。

参数

`query` *(optional enumerated Type array of string)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`callback` *(optional enumerated Type array of function)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

回调函数

回调函数格式如下：

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(array of BookmarkTreeNode results) {...};
```

`results` *(optional enumerated Type array of BookmarkTreeNode array of paramType paramType)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

update

void chrome.bookmarks.update(, string `id` , object `changes` , function `callback`)

Undocumented.

更新书签或者书签夹的属性。指定需要改变的属性；未指定的属性将不会被改变。注意： 近期只支持 'title'和 'url'。

参数

`id` *(optional enumerated Type array of string)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`changes` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`title` *(optional enumerated Type array of string)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`url` *(optional enumerated Type array of string)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`callback` *(optional enumerated Type array of function)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

回调函数

如果需要指定callback参数，函数格式如下：

```
function(BookmarkTreeNode result) {...};
```

`result` *(optional enumerated BookmarkTreeNode array of paramType)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

事件

onChanged

chrome.bookmarks.onChanged.addListener(function(string id, object changeInfo) {...});

Undocumented.

当书签或者书签夹发生改变时触发该事件。注意： 近期只有标题和url发生改变时，才触发该事件。

参数

`id` *(optional enumerated Type array of string)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`changeInfo` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`title` *(optional enumerated Type array of string)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`url` *(optional enumerated Type array of string)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

onChildrenReordered

chrome.bookmarks.onChildrenReordered.addListener(function(string id, object reorderInfo) {...});

Undocumented.

由于UI中的顺序发生改变时，书签夹会改变其子节点的顺序，此时会触发该事件。函数move()不会触发该事件。

Parameters

`id` *(optional enumerated Type array of string)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`reorderInfo` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`childIds` *(optional enumerated Type array of Type array of string paramType)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

onCreated

chrome.bookmarks.onCreated.addListener(function(string id, BookmarkTreeNode bookmark) {...});

Undocumented.

当创建书签或者书签夹夹时，会触发该事件。

参数

`id` *(optional enumerated Type array of string)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`bookmark` *(optional enumerated BookmarkTreeNode array of paramType)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

onImportBegan

chrome.bookmarks.onImportBegan.addListener(function() {...});

Undocumented.

当开始导入书签时，会触发该事件 。事件响应者在导入结束前不要处理标签创建、更新的事件。但仍然可以立即处理其他事件。

Parameters

onImportEnded

chrome.bookmarks.onImportEnded.addListener(function() {...});

Undocumented.

当导入书签结束时，会触发该事件 。

Parameters

onMoved

chrome.bookmarks.onMoved.addListener(function(string id, object moveInfo) {...});

Undocumented.

当书签或者书签夹被移动到其他父书签夹时，触发该事件。

参数

`id` *(optional enumerated Type array of string)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`moveInfo` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`parentId` *(optional enumerated Type array of string)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`index` *(optional enumerated Type array of integer)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`oldParentId` *(optional enumerated Type array of string)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`oldIndex` *(optional enumerated Type array of integer)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

onRemoved

chrome.bookmarks.onRemoved.addListener(function(string id, object removeInfo) {...});

Undocumented.

当书签和书签夹被删除时，触发该事件。当递归删除书签夹时，只会触发一个节点删除事件，它的子节点不会触发节点删除事件。

参数

`id` *(optional enumerated Type array of string)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`removeInfo` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`parentId` *(optional enumerated Type array of string)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`index` *(optional enumerated Type array of integer)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Types

BookmarkTreeNode

`paramName` *(optional enumerated Type array of object)*

Undocumented.

这个节点对象代表一个书签或者一个书签目录项。 节点对象有父子关系。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`id` *(optional enumerated Type array of string)*

Undocumented.

节点的唯一标识。IDs 在当前配置文件中是唯一的，浏览器重启后依然有效。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`parentId` (*optional enumerated Type array of string*)

Undocumented.

父节点的ID。根节点没有父节点。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`index` (*optional enumerated Type array of integer*)

Undocumented.

在父节点的书签夹范围内，该节点的索引，从0开始。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`url` (*optional enumerated Type array of string*)

Undocumented.

当用户点击书签时，浏览器访问的url。书签夹没有该属性 。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`title` (*optional enumerated Type array of string*)

Undocumented.

节点的说明文字。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`dateAdded` (*optional enumerated Type array of number*)

Undocumented.

节点创建时距纪元时间的毫秒数。（`new Date(dateAdded)`）.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`dateGroupModified` *(optional enumerated Type array of number)*

Undocumented.

书签夹内容的最后更新时间距纪元时间的毫秒数。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`children` *(optional enumerated Type array of BookmarkTreeNode array of paramType paramType)*

Undocumented.

节点的孩子的有序列表。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Cookies

内容

For information on how to use experimental APIs, see the chrome.experimental.* APIs page.

Cookies

清单

要使用cookies API, 你必须在你的清单中声明"cookies"权限，以及任何你希望cookie可以访问的主机权限。例如：

```
{
  "name": "My extension",
  ...
  **"permissions": [
    "cookies",
    "*://*.google.com"
  ]**,
  ...
}
```

范例

你可以在examples/api/cookies目录下找到一个使用cookies API的简单范例。对于其他的例子和查看源代码的帮助，参见 示例。

API 参考: chrome.cookies

属性

getLastError

chrome.extensionlastError

方法

get

void chrome.cookies.get(, object `details` , function `callback`)

Undocumented.

获取一个cookie的信息。如果对于给定的URL有多个cookie存在，将返回对应于最长路径的cookie。对于路径长度相同的cookies，将返回最早创建的cookie。

参数

`details` *(optional enumerated Type array of object对象)*

Undocumented.

用于识别所收到的cookie的详细信息。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`url` *(optional enumerated Type array of string字符串)*

Undocumented.

与所收到的cookie关联的URL。这个参数可以是一个完整的URL，这时候所有跟随在URL上的数据（比如查询字符串）将被忽略。如果清单文件中没有设置这个URL对应的主机权限，那么这个API调用会失败。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`name` *(optional enumerated Type array of string*字符串 *)*

Undocumented.

收到的cookie名字。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`storeId` *(optional enumerated Type array of string*可选，字符串 *)*

Undocumented.

cookie的存储id，用于从中检索cookie。默认情况下，当前执行上下文的cookie存储将被使用。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`callback` *(optional enumerated Type array of function* 函数 *)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

回调函数

如果指定了回调参数，它应该指定一个如下所示函数：

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(Cookie cookie) {...};
```

`cookie` *(optional enumerated Cookie array of paramType)*（可选，Cookie）

Undocumented.

包含cookie的详细信息。如果没找到cookie，该参数为null。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

getAll

void chrome.cookies.getAll(, object `details` , function `callback`)

Undocumented.

从一个cookie存储获取与给定信息匹配的所有cookies。所获取cookies将按照最长路径优先排序。当多个cookies有相同长度路径，最早创建的cookie在前面。

参数

`details` *(optional enumerated Type array of object)*

Undocumented.

对cookies进行筛选的信息。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`url` *(optional enumerated Type array of string可选，字符串)*

Undocumented.

限定只查找与给定URL匹配的cookies。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`name` (*optional enumerated Type array of string*可选，字符串)

Undocumented.

根据名称过滤cookies。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`domain` (*optional enumerated Type array of string*可选，字符串)

Undocumented.

限定只查找与给定域名或者子域名匹配的cookies。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`path` (*optional enumerated Type array of string*可选，字符串)

Undocumented.

限定只查找与给定路径完全一致的cookies。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`secure` (*optional enumerated Type array of boolean*可选，*Boolean*类型)

Undocumented.

根据cookie的Secure属性进行筛选。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`session` *(optional enumerated Type array of boolean可选， Boolean类型)*

Undocumented.

根据cookie的生命周期是会话的还是持久的进行过滤。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`storeId` *(optional enumerated Type array of string可选， 字符串)*

Undocumented.

cookie的存储id，用于从中检索cookie。默认情况下，当前执行上下文的cookie存储将被使用。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`callback` *(optional enumerated Type array of function)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

回调函数

如果指定了回调参数，它应该指定一个如下所示函数：

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(array of Cookie cookies) {...};
```

`cookies` *(optional enumerated Type array of Cookie array of paramType paramType 可选， cookie列表)*

Undocumented.

所有与给定cookie信息匹配、存在并未过期的cookie列表。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

getAllCookieStores

void chrome.cookies.getAllCookieStores(, function `callback`)

Undocumented.

列举所有存在的cookie存储。

参数

`callback` *(optional enumerated Type array of function)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

回调函数

如果指定了回调参数，它应该指定一个如下所示函数：

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(array of CookieStore cookieStores) {...};
```

`cookieStores` *(optional enumerated Type array of CookieStore* 可选，cookie存储列表 *array of paramType paramType)*

Undocumented.

所有存在的cookie存储。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

remove

void chrome.cookies.remove(, object `details`)

Undocumented.

根据名称删除cookie。

参数

`details` *(optional enumerated Type array of object)*

Undocumented.

用于鉴定待删除cookie的信息。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`url` *(optional enumerated Type array of string)*

Undocumented.

与所收到的cookie关联的URL。如果清单文件中没有设置这个URL对应的主机权限，那么这个API调用会失败。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`name` *(optional enumerated Type array of string)*

Undocumented.

待删除cookie的名称。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`storeId` *(optional enumerated Type array of string)*

Undocumented.

cookie的存储id，用于从中检索cookie。默认情况下，当前执行上下文的cookie存储将被使用。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

回调函数

The callback *parameter* should specify a function that looks like this:

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(Type param1, Type param2) {...};
```

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

set

void chrome.cookies.set(, object `details`)

Undocumented.

用给定数据设置一个cookie。如果相同的cookie存在，它们可能会被覆盖。

参数

`details` *(optional enumerated Type array of object)*

Undocumented.

待设置cookie的详细信息。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`url` *(optional enumerated Type array of string)*

Undocumented.

与待设置cookie相关的URL。该值影响所创建cookie的默认域名与路径值。如果清单文件中没有设置这个URL对应的主机权限，那么这个API调用会失败。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`name` *(optional enumerated Type array of string)*

Undocumented.

cookie名称，默认为空值。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`value` *(optional enumerated Type array of string)*

Undocumented.

cookie的值，默认为空值。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`domain` *(optional enumerated Type array of string)*

Undocumented.

cookie的域名。如果未指定，则该cookie是host-only cookie。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser

version.

`path` *(optional enumerated Type array of string)*

Undocumented.

cookie的路径。默认是url参数的路径部分。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`secure` *(optional enumerated Type array of boolean)*

Undocumented.

是否cookie标记为保密。默认为false。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`httpOnly` *(optional enumerated Type array of boolean)*

Undocumented.

是否cookie被标记为HttpOnly。默认为false。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`expirationDate` *(optional enumerated Type array of number)*

Undocumented.

cookie的过期时间，用从UNIX epoch开始计的秒数表示。如果未指定，该cookie是一个会话cookie。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`storeId` *(optional enumerated Type array of string)*

Undocumented.

用于保存该cookie的存储id。默认情况下，当前执行上下文的cookie存储将被使用。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

Callback function

The callback *parameter* should specify a function that looks like this:

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(Type param1, Type param2) {...};
```

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

事件

onChanged

chrome.cookies.onChanged.addListener(function(object changeInfo) {...});

Undocumented.

当一个cookie被设置或者删除时候触发。

参数

`changeInfo` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`removed` *(optional enumerated Type array of boolean)*

Undocumented.

True表示一个cookie被删除。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`cookie` *(optional enumerated Cookie array of paramType)*

Undocumented.

被设置或者删除的cookie的信息。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

类型

Cookie

`paramName` *(optional enumerated Type array of object)*

Undocumented.

表示一个HTTP cookie的信息。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`name` *(optional enumerated Type array of string)*

Undocumented.

cookie名称。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`value` *(optional enumerated Type array of string)*

Undocumented.

cookie值。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`domain` *(optional enumerated Type array of string)*

Undocumented.

cookie的域名。(例如 "www.google.com", "example.com").

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`hostOnly` *(optional enumerated Type array of boolean)*

Undocumented.

True表示cookie是一个host-only cookie (例如，一个检索的主机必须与cookie的域名完全一致）。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`path` *(optional enumerated Type array of string)*

Undocumented.

cookie的路径。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`secure` *(optional enumerated Type array of boolean)*

Undocumented.

True表示cookie被标记为保密。(例如，它的有效范围被限制于加密频道，最典型是HTTPS).

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`httpOnly` *(optional enumerated Type array of boolean)*

Undocumented.

True表示cookie被标记为HttpOnly (例如cookie在客户端的脚本无法访问)。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`session` *(optional enumerated Type array of boolean)*

Undocumented.

True表示cookie是线程cookie，与有过期时间的持久cookie相对应。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`expirationDate` *(optional enumerated Type array of number)*

Undocumented.

cookie的过期时间，用从UNIX epoch（00:00:00 UTC on 1 January 1970）开始计的秒数表示。会话cookie没有该属性。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`storeId` *(optional enumerated Type array of string)*

Undocumented.

包含该cookie的存储id，可通过getAllCookieStores()获取。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser

version.

CookieStore

`paramName` *(optional enumerated Type array of object)*

Undocumented.

表示浏览器中的cookie存储。那些隐藏模式窗体使用与非隐藏窗体不同的一个独立cookie存储。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`id` *(optional enumerated Type array of string)*

Undocumented.

cookie存储的唯一id。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`tabIds` *(optional enumerated Type array of Type array of integer paramType)*

Undocumented.

共享该cookie存储的浏览器标签集合的id列表。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

chrome.devtools.* APIs

下列API模块提供了开发人员工具的部分接口，以支持您对开发人员工具进行扩展。

- devtools.inspectedWindow
- devtools.network
- devtools.panels

如何使用 DevTools APIs

1. 1. 在扩展的manifest中指定 "devtools_page"项：

```
{
  "name": ...
  "version": "1.0",
  "minimum_chrome_version": "10.0",
  **"devtools_page": "devtools.html"**,
  ...
}
```

2. 2. 一旦开发人员工具窗口打开，将会创建一个 manifest 中 devtools_page 项指定的实例。通过使用 devtools.panels ， 这个页面可以在开发人员工具窗口中添加其他扩展页面，如面板或侧边栏等。

3. 3. chrome的devtools.* API模块仅适用于在开发人员工具窗口中加载的页面。Content scripts 和其他的扩展页面无法使用这些接口。也因此,该模块仅在开发人员工具窗口的生命周期内可用。
4. 4. 在开发人员工具窗口内扩展页面可用的接口包括所有的 上面列举出来的 devtools 模块 和 chrome.extension API。其他扩展API在开发人员工具页面都无法使用，您可以通过扩展的 background page 里发送请求来调用它们，这和在 content scripts 中调用类似。
5. 5. 还有一些开发人员工具API仍处于试验阶段。您可以通过 chrome.experimental.* APIs 获取这些API列表并了解如何使用它们。
6. 6. 意见反馈！ 您的意见和建议将会帮助我们改进这些API。

更多信息

想了解扩展可以使用的标准API信息,您可以访问 chrome.* APIs 和 Other APIs 。

示例

您可以通过 Samples 找到使用开发人员工具API的示例。

Events

For information on how to use experimental APIs, see the chrome.experimental.* APIs page.

`Event` 是一个对象，当你关注的一些事情发生时通知你。 以下是一个使用 `chrome.tabs.onCreated event` 的例子，每当一个新标签创建时，event对象会得到通知：

```
chrome.tabs.onCreated.**addListener(function(**tab**) {**
  appendToLog('tabs.onCreated --'
              + ' window: ' + tab.windowId
              + ' tab: '    + tab.id
              + ' index: '  + tab.index
              + ' url: '    + tab.url);
**});**
```

如示例所示，使用 `addListener()` 方法注册通知。 `addListener()` 方法的参数总是一个函数，是你定义来处理事件的函数， 但该函数的参数取决于你的事件处理。 查看 `chrome.tabs.onCreated` 的文档， 你可以看到该函数有一个参数：一个 Tab 对象，包含新创建的标签的信息。

方法

你可以调用任何 `Event` 对象的以下方法：

```
void addListener(function callback(...))
void removeListener(function callback(...))
bool hasListener(function callback(...))
```

chrome.history

chorme.history 模块被用于和浏览器所访问的页面记录交互。你可以添加、删除、查询浏览器的历史记录。如果您想覆写历史页面，请查看 Override替代页.

Manifest

您必须在扩展Manifest文件中定义"history"许可，以便使用history API. 如下所示：

```
{
  "name": "My extension",
  ...
  **"permissions": [
    "history"
  ]**,
  ...
}
```

过渡类型

History API用一种过渡类型来描述浏览器是如何访问的特定的URL。例如：如果URL是在用户访问页面时，点击链接跳转访问的，此时的过渡类型为 "link"。

如下的列表详细定义了各种过渡类型。.

过渡类型	描述
"link"	用户通过点击页面中的链接，跳转至本URL
"typed"	用户通过地址栏输入网址，来访问本URL。这种类型也适用于显式的导航动作。与之相反，你可以参阅generated，它适用于用户没看到（不知道）网址URL的情况。
"auto_bookmark"	用户通过界面的推荐到达本URL。例如，通过点击菜单项打开的页面。
"auto_subframe"	子框架导航。这种类型是指那些非顶层框架自动加载的内容。例如，如果一个页面由许多包含广告的子框架构成，那些广告链就拥有这种过渡类型。用户可能没有意识到页面中的这些内容是个单独的框架，所以他们也可能根本没有在意这些URL（请查阅manual_subframe)。
"manual_subframe"	此种类型是为用户显式请求的子框架导航以及在前进/后退列表中的生成导航入口的子框架导航所设置。由于用户更关心所请求框架被加载的效果，因此显式请求的框架可能会比自动载入的框架更为重要。
"generated"	用户通过在地址栏输入，而选择一个不像网址的入口到达的URL页面。例如，匹配结果中可能包含Google搜索结果页的URL, 但是它可能以"Google 搜索"的形式展现。这类导航和 typed 导航是有差异的，因为用户没有输入或者看到最终的URL。请参阅keyword。
"start_page"	页面是在命令行中被指定（打开），或者其本身就是起始页。
"form_submit"	用户提交的表单。请注意，某些情况，诸如表单运用脚本来提交，不属于此种类型。
"reload"	用户通过点击刷新按钮或者在地址栏输入回车键来刷新页面属于此种类型。会话重置，重开标签页都属于此种类型。
"keyword"	URL通过可替代的关键字，而不是默认的搜索引擎产生。请查阅keyword_generated 。
"keyword_generated"	相应由关键字生成的访问。请查阅 keyword。

示例

请查看 history 样例目录 和 history API 测试目录 中如何使用 API 的例子。 如果想获取源代码中的样例及帮助，请查看 样例。

API reference: chrome.history

Properties

getLastError

chrome.extensionlastError

Methods

addUrl

void chrome.history.addUrl(, object `details`)

Undocumented.

在当前历史记录中添加一条过渡类型为"link"的URL。

Parameters

`details` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`url` *(optional enumerated Type array of string)*

Undocumented.

要添加的URL。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

Callback function

The callback *parameter* should specify a function that looks like this:

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(Type param1, Type param2) {...};
```

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

deleteAll

void chrome.history.deleteAll(, function `callback`)

Undocumented.

删除历史记录中的所有条目。

Parameters

`callback` *(optional enumerated Type array of function)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

Callback function

回调函数的参数应当如下定义：

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function() {...};
```

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

deleteRange

void chrome.history.deleteRange(, object `range` , function `callback`)

Undocumented.

删除特定日期范围内的所有历史记录条目。页面本身只会在所有访问均在所设定日期的范围内才会被删除。

Parameters

`range` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`startTime` *(optional enumerated Type array of number)*

Undocumented.

条目添加起始时间， 以纪元开始的毫秒数表示。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`endTime` *(optional enumerated Type array of number)*

Undocumented.

条目添加终止时间， 以纪元开始的毫秒数表示。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`callback` *(optional enumerated Type array of function)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

Callback function

回调函数的参数应当如下定义：

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function() {...};
```

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

deleteUrl

void chrome.history.deleteUrl(, object `details`)

Undocumented.

删除指定URL 的所有历史记录。

Parameters

`details` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`url` *(optional enumerated Type array of string)*

Undocumented.

要删除的URL。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

Callback function

The callback *parameter* should specify a function that looks like this:

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(Type param1, Type param2) {...};
```

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

getVisits

void chrome.history.getVisits(, object `details` , function `callback`)

Undocumented.

获取访问特定URL的所有信息。

Parameters

`details` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`url` *(optional enumerated Type array of string)*

Undocumented.

需要获取访问信息的URL。URL 必须与历史记录搜索的返回值格式一致。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`callback` *(optional enumerated Type array of function)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

Callback function

回调函数的参数应当如下定义：

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(array of VisitItem results) {...};
```

`results` *(optional enumerated Type array of VisitItem array of paramType paramType)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

search

void chrome.history.search(, object `query` , function `callback`)

Undocumented.

搜索历史中，匹配条件的最后一次访问的页面。

Parameters

`query` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`text` *(optional enumerated Type array of string)*

Undocumented.

历史服务的文字查询。若想获取所有页面的信息，把此参数设置为空。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`startTime` *(optional enumerated Type array of number)*

Undocumented.

限制结果访问日期至此日期之后, 以纪元开始的毫秒数表示。.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`endTime` *(optional enumerated Type array of number)*

Undocumented.

限制结果访问日期至此日期以前, 以纪元开始的毫秒数表示。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`maxResults` *(optional enumerated Type array of integer)*

Undocumented.

所获取结果的最大数目。 默认值为100。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`callback` *(optional enumerated Type array of function)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

Callback function

回调函数的参数应当如下定义：

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(array of HistoryItem results) {...};
```

`results` *(optional enumerated Type array of HistoryItem array of paramType paramType)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Events

onVisitRemoved

chrome.history.onVisitRemoved.addListener(function(object removed) {...});

Undocumented.

当一条或着多条URL从历史服务中删除，此事件触发。当所有访问被移除后， 此条URL从来历史记录中被移除。

Parameters

`removed` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`allHistory` *(optional enumerated Type array of boolean)*

Undocumented.

如需删除所有历史记录，则设置成 True。如果设置成true, 所有url 为空。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`urls` *(optional enumerated Type array of Type array of string paramType)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

onVisited

chrome.history.onVisited.addListener(function(HistoryItem result) {...});

Undocumented.

当URL 被访问时，此事件被触发。并提供相应URL的HistoryItem数据。

Parameters

`result` *(optional enumerated HistoryItem array of paramType)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Types

HistoryItem

`paramName` *(optional enumerated Type array of object)*

Undocumented.

封装历史查询结果的对象。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`id` *(optional enumerated Type array of string)*

Undocumented.

条目的唯一标识符。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`url` *(optional enumerated Type array of string)*

Undocumented.

用户所访问的URL。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`title` *(optional enumerated Type array of string)*

Undocumented.

历史页面的标题。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`lastVisitTime` *(optional enumerated Type array of number)*

Undocumented.

页面被载入的最后时间，以纪元开始的毫秒数表示。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`visitCount` *(optional enumerated Type array of integer)*

Undocumented.

用户访问此页面的次数。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`typedCount` *(optional enumerated Type array of integer)*

Undocumented.

用户通过地址栏输入访问目的页面的次数。

VisitItem

`paramName` *(optional enumerated Type array of object)*

Undocumented.

封装URL访问的对象。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`id` *(optional enumerated Type array of string)*

Undocumented.

条目的唯一标识符。

`visitId` *(optional enumerated Type array of string)*

Undocumented.

访问的唯一标识符。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`visitTime` *(optional enumerated Type array of number)*

Undocumented.

访问的时间， 以纪元开始的毫秒数表示。

`referringVisitId` *(optional enumerated Type array of string)*

Undocumented.

访问来源的 visit_id 。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`transition` *(optional enumerated Type array of string ["link", "typed", "auto_bookmark", "auto_subframe", "manual_subframe", "generated", "start_page", "form_submit", "reload", "keyword", "keyword_generated"])*

Undocumented.

访问来源的过渡类型。

Management

`chrome.management` 模块提供了管理已安装和正在运行中的扩展或应用的方法。对于重写内建的新标签页的扩展尤其有用。

Manifest

要使用这个API，您必须在扩展清单文件中 中对授权，例如:

```
{
  "name": "My extension",
  ...
  **"permissions": [ "management" ]**,
  ...
}
```

只有一个方法不需要事先授权使用，那就是: `getPermissionWarningsByManifest`

API reference: chrome.management

方法

get

chrome.management.get(string `id` , function `callback`)

获得指定id的扩展/应用的信息。

参数

`id` *(string)*

应用或扩展（ExtensionInfo）中的ID。

`callback` *(optional function)*

回调

如果你指定了 回调函数 它看起来应该像下面这个样子：

```
function(ExtensionInfo result) {...};
```

`result` *(ExtensionInfo)*

getAll

chrome.management.getAll(function `callback`)

返回所有已安装的扩展。

参数

`callback` *(optional function)*

回调

如果你指定了 回调函数，它看起来应该像下面这个样子：

```
function(array of ExtensionInfo result) {...};
```

`result` *(array of ExtensionInfo)*

getPermissionWarningsById

chrome.management.getPermissionWarningsById(string `id` , function `callback`)

获得指定id的扩展的 授权 警告。

参数

`id` *(string)*

`callback` *(optional function)*

回调

如果你指定了 回调函数它看起来应该像下面这个样子：

```
function(array of string permissionWarnings) {...};
```

`permissionWarnings` *(array of string)*

getPermissionWarningsByManifest

chrome.management.getPermissionWarningsByManifest(string `manifestStr` , function `callback`)

获得指定的manifest清单文件中所包含的权限警告清单。注意，这一函数不需要在manifest文件中进行授权就可以使用。

参数

`manifestStr` *(string)*

扩展的manifest文件内容（是个JSON字符串）。

`callback` *(optional function)*

回调

如果你指定了 回调函数，它看起来应该像下面这个样子：

```
function(array of string permissionWarnings) {...};
```

`permissionWarnings` *(array of string)*

launchApp

chrome.management.launchApp(string `id` , function `callback`)

启动一个应用。

参数

`id` *(string)*

应用的唯一ID。

`callback` *(optional function)*

回调

如果你指定了 回调函数，它看起来应该像下面这个样子：

```
function() {...};
```

setEnabled

chrome.management.setEnabled(string `id` , boolean `enabled` , function `callback`)

启用或禁用一个应用或扩展。

参数

`id` *(string)*

应用或扩展（ExtensionInfo）的唯一ID。

`enabled` *(boolean)*

应用或扩展是否被启用或禁用。

`callback` *(optional function)*

回调

如果你指定了 回调函数，它看起来应该像下面这个样子：

```
function() {...};
```

uninstall

chrome.management.uninstall(string `id` , function `callback`)

卸载一个应用或扩展。

参数

`id` *(string)*

应用或扩展（ExtensionInfo）的唯一ID。

`callback` *(optional function)*

回调

If you specify the *callback* parameter, it should specify a function that looks like this:

如果你指定了 回调函数，它看起来应该像下面这个样子：

```
function() {...};
```

事件

onDisabled

chrome.management.onDisabled.addListener(function(ExtensionInfo info) {...});

当应用或扩展被禁用时触发。

参数

`info` *(ExtensionInfo)*

onEnabled

chrome.management.onEnabled.addListener(function(ExtensionInfo info) {...});

当应用或扩展被启用时触发。

参数

`info` (*[ExtensionInfo](#)*)

onInstalled

chrome.management.onInstalled.addListener(function(ExtensionInfo info) {...});

当应用或扩展被安装时触发。

参数

`info` (*[ExtensionInfo](#)*)

onUninstalled

chrome.management.onUninstalled.addListener(function(string id) {...});

当应用或扩展被卸载时触发。

参数

`id` (*string*)

被卸载的扩展或应用的id。

类型

IconInfo

(*object*)

扩展或应用的图标信息。

`size` (*integer*)

一个表示图标宽高的整数值，比如 128、48、24、16 或者其他值。

`url` (*string*)

该图标图像的URL。如果要显示一个灰度版本的图标（例如表示扩展程序已禁用时），请在URL后附加 `grayscale=true`

ExtensionInfo

(*object*)

已安装的扩展或应用的的信息。

`id` (*string*)

该扩展的唯一ID。

`name` *(string)*

扩展或应用的名字。

`description` *(string)*

扩展或应用的描述信息。

`version` *(string)*

扩展或应用的版本。

`mayDisable` *(boolean)*

该扩展是否允许用户禁用和卸载。

`enabled` *(boolean)*

该扩展当前是否被启用或禁用。

`disabledReason` *(optional enumerated string ["unknown", "permissions_increase"])*

当前扩展或应用被禁用的原因。

`isApp` *(boolean)*

是否是应用，如果true，则是。

`appLaunchUrl` *(optional string)*

应用的启动URL。

`homepageUrl` *(optional string)*

扩展或应用的主页。

`updateUrl` *(optional string)*

扩展或应用的升级页。

`offlineEnabled` *(boolean)*

扩展或应用是否支持离线使用。

`optionsUrl` *(string)*

扩展或应用的选项页，如果它们进行选项配置的话。

`icons` *(optional array of IconInfo)*

包含所有图标信息。注意这只反映声明在清单文件中的信息，URL指定的实际图像可能比声明的更大或更小，所以您引用这些图像时可能要考虑在图像标签中显式使用width和height属性。有关更多细节，请参见manifest documentation on icons。

`permissions` *(array of string)*

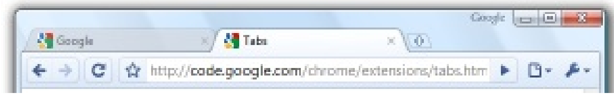
根据授权情况返回允许使用的所有API列表。

`hostPermissions` *(array of string)*

根据授权情况返回所有允许访问的主机白名单。

标签

chrome标签模块被用于和浏览器的标签系统交互。此模块被用于创建，修改，重新排列浏览器中的标签。.



Manifest

几乎所有chrome标签方法需要你在extension manifest中定义标签权限。例如：

```
{
  "name": "My extension",
  ...
  **"permissions": [
    "tabs"
  ]**,
  ...
}
```

不需要"标签"权限的方法是：create 和 update.

示例

你可以在examples/api/tabs 目录下找到运用标签模块的示例。在源代码中查看帮助或者示例，请查阅Samples

。

API reference: chrome.tabs

Properties

getLastError

chrome.extensionlastError

Methods

captureVisibleTab

void chrome.tabs.captureVisibleTab(, integer `windowId` , object `options` , function `callback`)

Undocumented.

在特定窗口中，抓取当前选中标签的可视区域。 这要求必须在 host permission 中指定对标签URL的访问权限。

Parameters

`windowId` *(optional enumerated Type array of integer)*

Undocumented.

目标窗口，默认值为当前窗口.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`options` *(optional enumerated Type array of object)*

Undocumented.

设置抓取图像参数。设置图像抓取的参数，比如生成的图像的格式。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`format` *(optional enumerated Type array of string ["jpeg", "png"])*

Undocumented.

生成的图像的格式。默认是jpeg。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`quality` *(optional enumerated Type array of integer)*

Undocumented.

如果格式是'jpeg'，控制结果图像的质量。此参数对PNG图像无效。当质量降低的时候，抓取的画质会下降，需要存储的字节数也会递减。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`callback` *(optional enumerated Type array of function)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

Callback function

回调参数应该如下定义：

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(string dataUrl) {...};
```

`dataUrl` *(optional enumerated Type array of string)*

Undocumented.

被抓取标签的可视区域的URL。此URL可能会作为图像的src属性。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

connect

Port chrome.tabs.connect(, integer `tabId` , object `connectInfo`)

Undocumented.

连接到特定标签中的content script(s)。 事件 chrome.extension.onConnect 将被触发给每个指定页面上运行的content script扩展。了解更多请查看 Content Script Messaging。

Parameters

`tabId` *(optional enumerated Type array of integer)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`connectInfo` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`name` *(optional enumerated Type array of string)*

Undocumented.

会被传输到监听连接事件的content scirpt的onConnect函数当中。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

`paramName` *(optional enumerated Port array of paramType)*

Undocumented.

可以和在指定标签中运行的内容脚本通信。当标签被关闭或者不存在时，端口的 onDisconnect 事件被触发。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Callback function

The callback *parameter* should specify a function that looks like this:

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(Type param1, Type param2) {...};
```

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

create

void chrome.tabs.create(, object `createProperties` , function `callback`)

Undocumented.

创建新的标签。注： 无需请求manifest的标签权限，此方法也可以被使用。

Parameters

`createProperties` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`windowId` *(optional enumerated Type array of integer)*

Undocumented.

创建新标签的目标窗口。默认是当前窗口 。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser

version.

`index` *(optional enumerated Type array of integer)*

Undocumented.

标签在窗口中的位置。 值在零至标签数量之间。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`url` *(optional enumerated Type array of string)*

Undocumented.

标签导航的初始页面。完整的URL 必须包含一个前缀 (如 'http://www.google.com', 不能写为 'www.google.com')。 相对 URL则与扩展所在的页面相对， 默认值为新标签页面。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`selected` *(optional enumerated Type array of boolean)*

Undocumented.

标签是否成为选中标签。默认为true。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`pinned` *(optional enumerated Type array of boolean)*

Undocumented.

标签是否被固定。默认值为false。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`callback` *(optional enumerated Type array of function)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

Callback function

The callback *parameter* should specify a function that looks like this:

回调 参数 应该如下定义：

```
function(Tab tab) {...};
```

`tab` *(optional enumerated Tab array of paramType)*

Undocumented.

所创建的标签的细节，包含新标签的ID。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

detectLanguage

void chrome.tabs.detectLanguage(, integer `tabId` , function `callback`)

Undocumented.

检测标签内容的主要语言。

Parameters

`tabId` *(optional enumerated Type array of integer)*

Undocumented.

默认为"当前窗口"所选定的标签。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`callback` *(optional enumerated Type array of function)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

Callback function

回调参数应该如下定义：

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(string language) {...};
```

`language` *(optional enumerated Type array of string)*

Undocumented.

ISO 语言编码，例如en 或者 fr。若要查看此方法支持的完整语言列表，请参阅kLanguageInfoTable 。第2列和第4列会被检测，而且第一个不为空的值会被返回。 简体中文是个例外，返回zh-CN。对于未知语言，返回und。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

executeScript

void chrome.tabs.executeScript(, integer `tabId` , object `details` , function `callback`)

Undocumented.

向页面注入JavaScript 脚本执行。要了解详情，请查阅内容脚本文档的 programmatic injection 部分。

Parameters

`tabId` *(optional enumerated Type array of integer)*

Undocumented.

运行脚本的标签ID；默认为当前窗口所选中的标签。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`details` *(optional enumerated Type array of object)*

Undocumented.

要执行的脚本内容，可选code或者file，但不能同时选两者。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`code` *(optional enumerated Type array of string)*

Undocumented.

要执行的脚本代码。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`file` *(optional enumerated Type array of string)*

Undocumented.

要执行的脚本文件。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`allFrames` *(optional enumerated Type array of boolean)*

Undocumented.

true的时候，给所有frame执行脚本。默认为false，只给顶级frame执行脚本。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`callback` *(optional enumerated Type array of function)*

Undocumented.

所有脚本执行后会被调用的回调。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

Callback function

The callback *parameter* should specify a function that looks like this:

回调 参数 应该如下定义：

```
function() {...};
```

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

get

void chrome.tabs.get(, integer `tabId` , function `callback`)

Undocumented.

获取指定标签的细节信息。

Parameters

`tabId` *(optional enumerated Type array of integer)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`callback` *(optional enumerated Type array of function)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

回调函数

回调 参数 应该如下定义：

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(Tab tab) {...};
```

`tab` *(optional enumerated Tab array of paramType)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

getAllInWindow

void chrome.tabs.getAllInWindow(, integer `windowId` , function `callback`)

Undocumented.

获取指定窗口所有标签的细节信息。

Parameters

`windowId` *(optional enumerated Type array of integer)*

Undocumented.

默认为当前窗口。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`callback` *(optional enumerated Type array of function)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

回调函数

T 回调 参数 应该如下定义：

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(array of Tab tabs) {...};
```

`tabs` *(optional enumerated Type array of Tab array of paramType paramType)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

getCurrent

void chrome.tabs.getCurrent(, function `callback`)

Undocumented.

获取生成脚本调用的标签。此函数不适用于脚本被非标签内容调用的情况。(例如: 背景页 或者 弹出视图) 。

Parameters

`callback` *(optional enumerated Type array of function)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

回调函数

回调 参数 应该如下定义 :

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(Tab tab) {...};
```

`tab` *(optional enumerated Tab array of paramType)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

getSelected

void chrome.tabs.getSelected(, integer `windowId` , function `callback`)

Undocumented.

获取特定窗口指定的标签。

Parameters

`windowId` *(optional enumerated Type array of integer)*

Undocumented.

默认为当前窗口。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`callback` *(optional enumerated Type array of function)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

回调函数

回调 参数 应该如下定义：

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(Tab tab) {...};
```

`tab` *(optional enumerated Tab array of paramType)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

insertCSS

void chrome.tabs.insertCSS(, integer `tabId` , object `details` , function `callback`)

Undocumented.

向页面注入CSS。要了解详情，请参阅内容脚本文档中的 programmatic injection 部分。

Parameters

`tabId` *(optional enumerated Type array of integer)*

Undocumented.

要注入CSS的标签ID；默认为当前窗口选定的标签。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`details` *(optional enumerated Type array of object)*

Undocumented.

要注入的CSS的内容，可选code或者file，但不能同时选两者。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`code` *(optional enumerated Type array of string)*

Undocumented.

要注入的CSS代码。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`file` *(optional enumerated Type array of string)*

Undocumented.

要注入的CSS文件。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`allFrames` *(optional enumerated Type array of boolean)*

Undocumented.

true的时候，给所有frame注入CSS。默认为false，只给顶级frame注入CSS。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`callback` *(optional enumerated Type array of function)*

Undocumented.

当所有的CSS 被注入后，回调被调用。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

回调函数

回调 参数 应该如下定义：

```
function() {...};
```

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

move

void chrome.tabs.move(, integer `tabId` , object `moveProperties` , function `callback`)

Undocumented.

把标签移动至窗口内特定的位置，或者移至一个新窗口。请注意只能在普通窗口之间切移 (window.type === "normal") 。

Parameters

`tabId` *(optional enumerated Type array of integer)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`moveProperties` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`windowId` *(optional enumerated Type array of integer)*

Undocumented.

默认为标签所在的窗口。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`index` *(optional enumerated Type array of integer)*

Undocumented.

移动到的目标窗口位置。赋值必须在零至目标窗口的标签数目之间。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`callback` *(optional enumerated Type array of function)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

回调函数

回调 参数 应该如下定义：

```
function(Tab tab) {...};
```

`tab` *(optional enumerated Tab array of paramType)*

Undocumented.

所被移动的标签细节。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

remove

void chrome.tabs.remove(, integer `tabId` , function `callback`)

Undocumented.

关闭标签。

Parameters

`tabId` *(optional enumerated Type array of integer)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`callback` *(optional enumerated Type array of function)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

回调函数

回调 参数 应该如下定义：

```
function() {...};
```

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

sendRequest

void chrome.tabs.sendRequest(, integer `tabId` , any `request` , function `responseCallback`)

Undocumented.

向特定的标签content script发送一个的请求， 并在响应返回时，可附带一个回调。在所有content script响应请求后， chrome.extension.onRequest 事件将会为当前扩展触发。

Parameters

`tabId` *(optional enumerated Type array of integer)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`request` *(optional enumerated Type array of any)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`responseCallback` *(optional enumerated Type array of function)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Parameters

`response` *(optional enumerated Type array of any)*

Undocumented.

响应的JSON对象。如果错误发生，回调将不会有参数。并会在 chrome.extension.lastError 产生一个错误。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

回调函数

回调 参数 应该如下定义：

```
function(any response) {...};
```

`response` *(optional enumerated Type array of any)*

Undocumented.

响应的JSON对象。如果错误发生，回调将不会有参数。并会在 chrome.extension.lastError 产生一个错误。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

update

void chrome.tabs.update(, integer `tabId` , object `updateProperties` , function `callback`)

Undocumented.

修改标签的属性。没有在updateProperties 中指定的属性不会被修改。注：即使没有向manifest 请求'tabs'权限，这个方法依然适用。

Parameters

`tabId` *(optional enumerated Type array of integer)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`updateProperties` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`url` *(optional enumerated Type array of undefined)*

Undocumented.

让标签浏览的URL。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`selected` *(optional enumerated Type array of boolean)*

Undocumented.

标签是否应被选中。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`pinned` *(optional enumerated Type array of boolean)*

Undocumented.

标签是否应被固定。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`callback` *(optional enumerated Type array of function)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

回调函数

The callback *parameter* should specify a function that looks like this:

回调 参数 应该如下定义：

```
function(Tab tab) {...};
```

`tab` *(optional enumerated Tab array of paramType)*

Undocumented.

被更新的标签细节。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Events

onAttached

chrome.tabs.onAttached.addListener(function(integer tabId, object attachInfo) {...});

Undocumented.

当标签附着在窗口上，此事件被触发。例如，此事件会发生在标签在窗口之前移动时。

Parameters

`tabId` *(optional enumerated Type array of integer)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`attachInfo` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`newWindowId` *(optional enumerated Type array of integer)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`newPosition` *(optional enumerated Type array of integer)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

onCreated

chrome.tabs.onCreated.addListener(function(Tab tab) {...});

Undocumented.

标签创建时，此事件触发。请注意，当事件触发时，标签的 URL 可能没有被设置， 但是当URL被设置时，可以通过onUpdated 事件接听。

Parameters

`tab` *(optional enumerated Tab array of paramType)*

Undocumented.

标签创建的细节。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

onDetached

chrome.tabs.onDetached.addListener(function(integer tabId, object detachInfo) {...});

Undocumented.

当标签从窗口脱离时，此事件被触发，例如标签在窗口之间移动。

Parameters

`tabId` *(optional enumerated Type array of integer)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`detachInfo` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`oldWindowId` *(optional enumerated Type array of integer)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`oldPosition` *(optional enumerated Type array of integer)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

onMoved

chrome.tabs.onMoved.addListener(function(integer tabId, object moveInfo) {...});

Undocumented.

当标签在窗口内移动时，此事件被触发。只有一个移动事件被触发，给用户直接移动的标签。其他响应移动事件的标签不触发移动事件。 请参阅 onDetached.查看详情。

Parameters

`tabId` *(optional enumerated Type array of integer)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`moveInfo` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`windowId` *(optional enumerated Type array of integer)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`fromIndex` *(optional enumerated Type array of integer)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`toIndex` *(optional enumerated Type array of integer)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

onRemoved

chrome.tabs.onRemoved.addListener(function(integer tabId, object removeInfo) {...});

Undocumented.

标签关闭时被触发。

Parameters

`tabId` *(optional enumerated Type array of integer)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`removeInfo` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`isWindowClosing` *(optional enumerated Type array of boolean)*

Undocumented.

当窗口被关闭，标签随之被关闭时，此参数为true。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

onSelectionChanged

chrome.tabs.onSelectionChanged.addListener(function(integer tabId, object selectInfo) {...});

Undocumented.

当窗口选中的标签改变时，此事件触发

Parameters

`tabId` *(optional enumerated Type array of integer)*

Undocumented.

被选中标签的ID。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`selectInfo` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`windowId` *(optional enumerated Type array of integer)*

Undocumented.

标签发生变化的窗口ID。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

onUpdated

chrome.tabs.onUpdated.addListener(function(integer tabId, object changeInfo, Tab tab) {...});

Undocumented.

当标签更新时，此事件被触发。

Parameters

`tabId` *(optional enumerated Type array of integer)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`changeInfo` *(optional enumerated Type array of object)*

Undocumented.

列出标签更新时的状态。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`status` *(optional enumerated Type array of string)*

Undocumented.

标签的状态。可以为 *loading* or *complete*。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`url` *(optional enumerated Type array of string)*

Undocumented.

经历变化的标签的URL。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`pinned` *(optional enumerated Type array of boolean)*

Undocumented.

标签被锁定的新状态。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`tab` *(optional enumerated Tab array of paramType)*

Undocumented.

更新的标签的状态。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Types

Tab

`paramName` *(optional enumerated Type array of object)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`id` *(optional enumerated Type array of integer)*

Undocumented.

标签ID。在一个浏览器会话内， 标签ID是唯一的。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`index` *(optional enumerated Type array of integer)*

Undocumented.

窗口内从零开始的标签索引。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`windowId` *(optional enumerated Type array of integer)*

Undocumented.

标签所在窗口的窗口ID。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`selected` *(optional enumerated Type array of boolean)*

Undocumented.

标签是否被选中。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`pinned` *(optional enumerated Type array of boolean)*

Undocumented.

标签是否被锁定。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`url` *(optional enumerated Type array of string)*

Undocumented.

标签所显示的URL。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`title` *(optional enumerated Type array of string)*

Undocumented.

标签的标题。当标签被加载时，标题可能不能被成功获取。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`favIconUrl` *(optional enumerated Type array of string)*

Undocumented.

标签收藏夹图标的URL。当标签被加载时，图标可能不能被成功获取。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`status` *(optional enumerated Type array of string)*

Undocumented.

可以被设置为 *loading* 或者 *complete*。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`incognito` *(optional enumerated Type array of boolean)*

Undocumented.

可以被设置为 *loading* 或者 *complete*.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

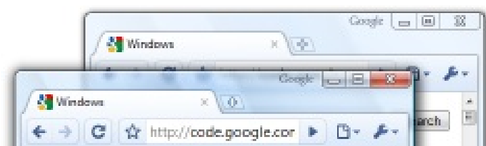
视窗

内容

For information on how to use experimental APIs, see the chrome.experimental.* APIs page.

视窗

使用chrome.windows模块与浏览器视窗进行交互。 你可以使用这个模块在浏览器中创建、修改和重新排列视窗。



清单

要使用视窗 API，你必须在manifest.json声明"tabs"的权限 。（不，这不是一个错字 - 窗口和标签模块的互动如此密切，我们决定它们共享一个权限。）例如：

```
{
  "name": "My extension",
  ...
  **"permissions": ["tabs"]**,
  ...
}
```

当前视窗

很多扩展系统的功能有一个可选的windowId参数，其默认值为当前视窗。

当前视窗是指包含当前正在执行的代码的视窗。重要的是要认识到，它可以跟最顶层或有焦点的视窗不一样。

例如，假设一个扩展从一个单一的HTML文件中创建了一些标签或视窗，而这个HTML文件包含一个chrome.tabs.getSelected的调用 。 当前视窗是指那个包含了发起调用的页面的视窗，不管它是不是最顶层视窗。

在背景页这个例子中 ，当前视窗就是最后一个活动视窗。在某些情况下，背景页可能没有当前视窗。

范例

你可以在examples/api/windows 目录下找到一些使用windows模块的简单范例。另外一个范例是在tabs_api.html文件中的检查器范例。 对于其他的例子和查看源代码的帮助，参见示例。

API 参考: chrome.windows

属性

WINDOW_ID_NONE

chrome.windows.WINDOW_ID_NONE

`WINDOW_ID_NONE` *(optional enumerated Type array of 整数)*

Undocumented.

这个windowId值表示没有Chrome浏览器窗口的情况。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

方法

create

void chrome.windows.create(, object `createData` , function `callback`)

Undocumented.

使用任何可选大小、位置或者默认提供的URL来创建（打开）一个新的浏览器。

参数

`createData` *(optional enumerated Type array of object*可选，对象 *)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`url` *(optional enumerated Type array of string or array of string* 可选，字符串或者字符串数组*)*

Undocumented.

一个或者一组在视窗作为标签打开的URL。完全合格的URL必须包括一个类型（即'http://www.google.com'，不是'www.google.com'）。相对URL将与扩展内的当前页相关。默认为新的标签页面。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`tabId` *(optional enumerated Type array of integer*可选，整数 *)*

Undocumented.

你想要在新视窗选定的标签的id。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`left` *(optional enumerated Type array of integer* 可选，整数 *)*

Undocumented.

新视窗相对于屏幕的左边缘的位置的像素值。如果没有指定，那么新的视窗从最后一个有焦点的视窗自然偏移。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`top` *(optional enumerated Type array of integer*可选，整数 *)*

Undocumented.

新视窗相对于屏幕的上边缘的位置的像素值。如果没有指定，那么新的视窗从最后一个有焦点的视窗自然偏移。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`width` *(optional enumerated Type array of integer*可选，整数 *)*

Undocumented.

新视窗的像素宽度。如果没有指定，默认为自然宽度。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`height` *(optional enumerated Type array of integer*可选，整数 *)*

Undocumented.

新视窗的像素高度。如果没有指定，默认为自然高度。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`incognito` *(optional enumerated Type array of boolean可选， Boolean类型)*

Undocumented.

新视窗是否是隐身。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`type` *(optional enumerated Type array of string ["normal", "popup"]可选，枚举字符串 ["normal", "popup"])*

Undocumented.

指定新建浏览器视窗的类型。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`callback` *(optional enumerated Type array of function可选，函数)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

回调函数

The callback *parameter* should specify a function that looks like this:

如果指定了回调参数，它应该指定一个如下所示函数：

```
function(Window window) {...};
```

`window` *(optional enumerated Window 可选，视窗 array of paramType)*

Undocumented.

包含新创建视窗的详细信息。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

get

void chrome.windows.get(, integer `windowId` , function `callback`)

Undocumented.

获取有关窗口的详细信息。

参数

`windowId` *(optional enumerated Type array of integer整数)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`callback` *(optional enumerated Type array of function函数)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser

version.

Returns

Callback function

如果指定了回调参数，它应该指定一个如下所示函数：

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(Window window) {...};
```

`window` *(optional enumerated Window array of paramType)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

getAll

void chrome.windows.getAll(, object `getInfo` , function `callback`)

Undocumented.

获得所有的视窗。

参数

`getInfo` *(optional enumerated Type array of object可选，对象)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`populate` *(optional enumerated Type array of boolean 可选，Boolean类型)*

Undocumented.

如果是true表示每个视窗对象都有一个包含该视窗所有标签的tabs属性。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`callback` *(optional enumerated Type array of function函数)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

回调函数

回调参数应该指定一个如下函数：

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(array of Window windows) {...};
```

`windows` *(optional enumerated Type array of Window array of paramType paramType)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

getCurrent

void chrome.windows.getCurrent(, function `callback`)

Undocumented.

获得当前视窗。

参数

`callback` *(optional enumerated Type array of function 函数)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

回调函数

如果指定了回调参数，它应该指定一个如下所示函数：

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(Window window) {...};
```

`window` *(optional enumerated Window array of paramType)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

getLastFocused

void chrome.windows.getLastFocused(, function `callback`)

Undocumented.

获取最近有焦点的视窗 — 一般是最顶层的视窗。

参数

`callback` *(optional enumerated Type array of function 函数)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

回调函数

如果指定了回调参数，它应该指定一个如下所示函数：

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(Window window) {...};
```

`window` *(optional enumerated Window array of paramType)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

remove

void chrome.windows.remove(, integer `windowId` , function `callback`)

Undocumented.

关闭一个视窗以及其包含的所有标签。

参数

`windowId` *(optional enumerated Type array of integer整数)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`callback` *(optional enumerated Type array of function 可选，函数)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

回调函数

The callback *parameter* should specify a function that looks like this:

如果指定了回调参数，它应该指定一个如下所示函数：

```
function() {...};
```

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

update

void chrome.windows.update(, integer `windowId` , object `updateInfo` , function `callback`)

Undocumented.

更新一个视窗的属性。只指定那些你希望修改的属性，未指定的属性将保持不变。

参数

`windowId` *(optional enumerated Type array of integer整数)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`updateInfo` *(optional enumerated Type array of object对象)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`left` *(optional enumerated Type array of integer 可选，整数)*

Undocumented.

视窗相对屏幕左边界进行移动的像素偏移值。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`top` *(optional enumerated Type array of integer 可选，整数)*

Undocumented.

视窗相对屏幕上边界进行移动的像素偏移值。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`width` *(optional enumerated Type array of integer可选，整数)*

Undocumented.

视窗宽度调整的像素值。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`height` *(optional enumerated Type array of integer可选，整数)*

Undocumented.

视窗高度调整的像素值。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`focused` *(optional enumerated Type array of boolean可选，Boolean类型)*

Undocumented.

如果是true，将该视窗提至前面。否则，将z-order上下一视窗提至前面。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`callback` *(optional enumerated Type array of function 可选，函数)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Returns

回调函数

The callback *parameter* should specify a function that looks like this:

如果指定了回调参数，它应该指定一个如下所示函数：

```
function(Window window) {...};
```

`window` *(optional enumerated Window array of paramType)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

事件

onCreated

chrome.windows.onCreated.addListener(function(Window window) {...});

Undocumented.

当一个新视窗被创建时触发。

参数

`window` *(optional enumerated Window array of paramType)*

Undocumented.

被创建窗体的详细信息。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

onFocusChanged

chrome.windows.onFocusChanged.addListener(function(integer windowId) {...});

Undocumented.

当前获得焦点窗口改变时触发。Will be chrome.windows.WINDOW_ID_NONE if all chrome windows have lost focus. Note: On some Linux window managers, WINDOW_ID_NONE will always be sent immediately preceding a switch from one chrome window to another.

Parameters

`windowId` *(optional enumerated Type array of integer)*

Undocumented.

ID of the newly focused window.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

onRemoved

chrome.windows.onRemoved.addListener(function(integer windowId) {...});

Undocumented.

当一个视窗被关闭时触发。

参数

`windowId` *(optional enumerated Type array of integer 整数)*

Undocumented.

被关闭视窗的ID。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

类型

Window

`paramName` *(optional enumerated Type array of object视窗，对象)*

Undocumented.

Description of this parameter from the json schema.

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`id` *(optional enumerated Type array of integer 整数)*

Undocumented.

视窗的ID。视窗ID在一个浏览器会话中是唯一的。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`focused` *(optional enumerated Type array of boolean)*

Undocumented.

该窗口是否当前焦点窗口。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`top` *(optional enumerated Type array of integer整数)*

Undocumented.

窗体相对屏幕上边缘的偏移像素值。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`left` *(optional enumerated Type array of integer整数)*

Undocumented.

窗体相对屏幕左边缘的偏移像素值。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`width` *(optional enumerated Type array of integer整数)*

Undocumented.

窗体宽度像素值。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`height` *(optional enumerated Type array of integer整数)*

Undocumented.

窗体高度像素值。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`tabs` *(optional enumerated Type array of Tab*可选，标签数组 *array of paramType paramType)*

Undocumented.

表征窗体所包含标签的数组。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`incognito` *(optional enumerated Type array of boolean)*

Undocumented.

窗体是否隐藏。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

`type` *(optional enumerated Type array of string ["normal", "popup", "app"])*

Undocumented.

浏览器视窗类型。

This parameter was added in version . You must omit this parameter in earlier versions, and you may omit it in any version. If you require this parameter, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

实现扩展

无障碍性(a11y)

当你设计一款扩展，需要让扩展对于诸如视觉缺陷，失聪，行动不便的残疾人没有使用障碍。

所有人 — 不仅仅是有特殊需求的人 — 都应该能从那些无障碍扩展所提供的相应模式中获益。例如，键盘的快捷键对于盲人，灵敏度较差的那些人非常重要，然而他们也能提高高级用户在无鼠标状态下的工作效率。 字幕和手抄本提供了聋人获取影音内容的通道，然而他们对语言学习者也非常有用。

人们可以通过各种方式和扩展交互。他们可能使用标准的显示器、键盘、鼠标，他们也可能使用显示放大器和键盘。另外还有一种情况是屏幕阅读机，一种为盲人和有视觉缺陷的人解释屏幕内容的辅助工具。屏幕阅读机可以大声朗读或者输出盲文。

虽然你不能预测人们会使用何种工具，但通过一系列的指引，你可以写出一款对大多数人无障碍的扩展。虽然本篇的指引不能让你的扩展对每一个人都没有使用障碍，不过这些指引确实是个好的开始。

使用无障碍的UI控件

首先，请使用支持无使用障碍的UI控件。最便捷的获取无障碍控件的方法是用标准HTML控件。如果你要创建自定义控件，请记住从设计之初就考虑控件的无障碍性，要比完工之后再支持无障碍性容易的多。

标准控件

在任何时候都尝试使用标准的 HTML UI 控件。标准的HTML控件 (如以下图例所示) 支持键盘无障碍性，可伸缩性，而且能被屏幕阅读机所理解。

Button	<input type="button" value="Button" />
Checkbox	<label for="check1">Checkbox</label> <input type="checkbox" id="check1" />
Radio button	<label for="radio1">Radio button</label> <input type="radio" id="radio1" />
Text field	<input type="text" value="Text field" />
Option 1 ▼	<select> <option value="Option 1">Option 1</option> <option value="Option 2">Option 2</option> <option value="Option 3">Option 3</option> </select>
Link	Link

自定义控件中的ARIA

ARIA 是一份通过一系列标准 DOM 属性使UI控件能被屏幕阅读器接入的规范。这些属性为屏幕阅读器提供了功能和当前页面控件状态的线索。 ARIA 是 W3C进行过程中的项目。

为插件中的自定义控件添加ARIA支持包含修改 DOM 属性， 为Google Chrome添加用户交互时引发事件的属性。屏幕阅读器相应这些事件并描述控件的功能。ARIA定义的DOM属性分为角色， 状态， 和 属性。

ARIA 角色 属标明了控件类型并描述了其行为。它表现为DOM属性角色，并附带一组预定义的ARIA 角色值。因为ARIA 角色是静态的，角色属性不应该改变它的值。

ARIA 角色规范 包含了如何挑选正确角色的细节信息。例如，如果你的插件包含工具条，工具栏DOM元素的角色属性应该如下设置：

```
<div role="toolbar">
```

ARIA 属性也用于描述一类特殊角色控件的当前状态和属性。状态 是动态的而且应当在用户交互时被更新。例如，角色为 "checkbox"的控件可以为 "checked" or "unchecked". 属性 总体上来说不是动态的，不过和描述控件特殊信息的状态相似。想了解ARIA 状态和属性的更多信息， 请参见W3C States and Properties specification.

注释: 你不需要为一种特定角色使用所有的状态和属性。

如下是一个添加ARIA 属性 aria-activedescendant 到工具栏控件的例子:

```
<div role="toolbar" tabindex="0" aria-activedescendant="button1">
```

aria-activedescendant 属性定义了当工具栏获取焦点时，哪一个工具栏的子控件获取了焦点。在此例中，工具栏的第一个控件（拥有id "button1"）是能获取焦点的子控件。代码tabindex="0" 定义了工具栏中的子控件获取焦点的DOM文档顺序。

如下是样例工具栏的完整定义:

```
<div role="toolbar" tabindex="0" aria-activedescendant="button1">
  <img src="buttoncut.png" role="button" alt="cut" id="button1">
  <img src="buttoncopy.png" role="button" alt="copy" id="button2">
  <img src="buttonpaste.png" role="button" alt="paste" id="button3">
</div>
```

一旦 ARIA 角色，状态，属性添加到了控件的DOM节点，Google Chrome 触发了合适的事件到屏幕阅读器。因为ARIA 支持还是一项正在进行中的工作，Google Chrome 可能不会触发ARIA属性的所有事件，而屏幕阅读器也可能不能识别所有被触发的事件。你可以在Chromium Accessibility Design Document 获取Google Chrome在ARIA支持上的信息。

如想获取添加ARIA控件到自定义控件的快速指南，请参见 Dave Raggett's presentation from WWW2010 。

自定义控件的焦点信息

请确保你扩展的操作和导航控件可以接受键盘焦点。操作控件可能包含按钮，树，列表框。导航控件包含标签和菜单栏。

默认来说，能在 HTML DOM 中接受键盘焦点的控件是锚，按钮和表单空间。然而，设置HTML属性 tabIndex 为 0，把DOM 属性放置到默认的标签序列，使他们能接受键盘焦点。 例如:

```
_element_.tabIndex = 0
```

设置 tabIndex = -1 从tab 序列中移出了元素，但是语法上仍然允许元素接受键盘焦点。如下是设置键盘焦点的实例：

```
_element_.focus();
```

请保证你的自定义控件包含键盘的支持，不仅对不使用鼠标用户非常重要，而且屏幕阅读器通过键盘来决定去描述哪个控件。

键盘入口的支持

用户在他们不想使用鼠标的时候也应该能使用你的扩展。

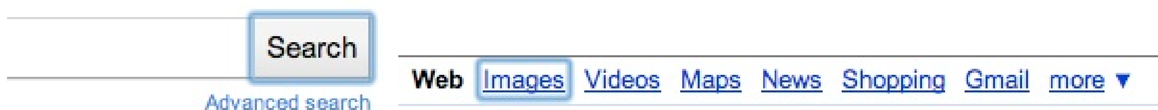
导航

请确认用户可以不用鼠标在你扩展的不同部分导航。也请确认任何页面或者浏览器的弹出是可以通过键盘导航的。.

在Windows上，可以使用**Shift+Alt+T** 切换键盘焦点至工具栏，这样就可以前往页面动作和浏览器动作的图标。有关于 键盘和鼠标 的帮助主题罗列出了Google Chrome所有的键盘快捷键，其中在 **Google Chrome** 功能快捷键 中也介绍了工具栏导航的诸多细节。

注释: Google Chrome 6 的 Windows 版本是第一个支持工具栏键盘导航的版本。Linux的支持也在计划中。在 Mac OS X上，Voice Over，Apple 屏幕阅读器提供了工具栏的接入。

请确保有键盘焦点的用户界面是易于辨认的。通常情况下在界面周围有焦点的外框，但是如果CSS被重度使用，外框线将被抑制，对比度可能被减弱。下面是焦点外框的两个例子。



快捷键

尽管最常见的快捷键导航提供了用TAB键在扩展界面中切换焦点的策略，然而这不是使用用户界面最简单高效的做法。如果提供了显示的键盘快捷键，键盘导航会变得更便捷。

如果要实现快捷键，控件上需要添加键盘响应。DHTML 风格指南工作组的 键盘快捷键指南 是一份很好的参考。

一个保证键盘快捷键易察觉性的方法是把他们罗列出来。扩展的 Options page 很适合完成此类事物。.

比如说工具栏，简单的JS键盘处理器可能如下所示。请注意为了反映当前的激活工具栏按钮，在响应用户输入后ARIA aria-activedescendant 属性是如何更新的。

```
<head>
<script>
 function optionKeyEvent(event) {
  var tb = event.target;
  var buttonid;
  ENTER_KEYCODE = 13;
  RIGHT_KEYCODE = 39;
  LEFT_KEYCODE = 37;
  // Partial sample code for processing arrow keys.
  if (event.type == "keydown") {
    // Implement circular keyboard navigation within the toolbar buttons
    if (event.keyCode == ENTER_KEYCODE) {
      ExecuteButtonAction(getCurrentButtonID());
      _// getCurrentButtonID defined elsewhere_
    } else if (event.keyCode == event.RIGHT_KEYCODE) {
      // Change the active toolbar button to the one to the right (circular).
      var buttonid = getNextButtonID();
      _// getNextButtonID defined elsewhere_
      tb.setAttribute("aria-activedescendant", buttonid);
    } else if (event.keyCode == event.LEFT_KEYCODE) {
      // Change the active toolbar button to the one to the left (circular).
      var buttonid = getPrevButtonID();
      _// getPrevButtonID defined elsewhere_
      tb.setAttribute("aria-activedescendant", buttonid);
    } else {
      return true;
    }
    return false;
  }
}
</script>
<div role="toolbar" tabindex="0" aria-activedescendant="button1" id="tb1"
     onkeydown="return optionKeyEvent(event);"
     onkeypress="return optionKeyEvent(event);">
  <img src="buttoncut" role="button" alt="cut" id="button1">
  <img src="buttoncopy" role="button" alt="copy" id="button1">
  <img src="buttonpaste" role="button" alt="paste" id="button1">
</div>
```

提供无障碍的内容

以下的指引可能为大众所熟知，因为他们反应了所有网页内容好的经验，不仅仅是扩展。

文字

评估你扩展中的文字。很多人可能发现在扩展中增加字体的大小会非常适用。如果你使用键盘快捷键，请确保他们不和Google Chrome自带的缩放快捷键冲突。

200% 测试可以作为你UI 灵活性的指示器。如果你增大字符大小或者放大页面至200%，你的扩展是否还有用？

请尽量避免在图像中夹杂文字: 用户不能改变以图像方式显示的文字大小，屏幕阅读器也不能解读图像。考虑适用网络字体，比如Google Font API 所提供的字体。 网络字体中的文字可供搜索，能缩放自如，并对屏幕阅读机的用户也是无障碍的。

颜色

请确保背景色和你扩展中的前景色/ 文字颜色有鲜明的对比度。 对比度测试工具 能检测到你的背景色和前景色是否提供了合适的对比度。如果你在Windows 环境中开发，你可以使用高对比度模式来检查你扩展的对比度。 当检测对比度时，请确认你扩展中依靠颜色、图像来传输信息的部分是容易分辨的。对于一些特定的图像，你可以使用诸如Vischeck simulation tool 的实用工具来检查在不同颜色缺陷下，图像的显示情况。

你可以考虑提供多种颜色主题，或者给用户自定义颜色主题来改善对比度的权利。

声音

如果扩展依赖影音来传输信息，请确保提供相应的字幕和副本。请参见 Described and Captioned Media Program guidelines 以获取有关于字幕的更多信息。

图像

请为你的图像提供信息化的可替代文本。如下所示:

```
<img src="img.jpg" alt="The logo for the extension">
```

尽量用可替代的文字表明图像的意图，这比繁复描述图像内容好的多。间隔图像或者纯装饰图像应该用（""）的可替代问题，或者HTML 和 css中移除。

如果必须在图像中使用文字，在可替代文字中添加图像文字。WebAIM article on appropriate alt text 是个好的参考。

示例

想查看实现键盘导航和ARIA属性的例子，请参见examples/extensions/news_a11y (与 examples/extensions/news 比较） 是实现键盘导航和ARIA属性的好的例子。想查看源代码中更多的例子，请参见 Samples。

背景页

扩展常常用一个单独的长时间运行的脚本来管理一些任务或者状态。 Background pages to the rescue.

如同 architecture overview 的解释。背景页是一个运行在扩展进程中的HTML页面。它在你的扩展的整个生命周期都存在，同时，在同一时间只有一个实例处于活动状态。

在一个有背景页的典型扩展中，用户界面（比如，浏览器行为或者页面行为和任何选项页）是由沉默视图实现的。当视图需要一些状态，它从背景页获取该状态。当背景页发现了状态改变，它会通知视图进行更新。

清单

请在扩展清单中注册背景页。一般，背景页不需要任何HTML，仅仅需要js文件，比如：

```
{
  "name": "My extension",
  ...
  **"background": {
    "scripts": ["background.js"]
  }**,
  ...
}
```

浏览器的扩展系统会自动根据上面 `scripts` 字段指定的所有js文件自动生成背景页。

如果您的确需要自己的背景页，可以使用 `page` 字段，比如：

```
{
  "name": "My extension",
  ...
  **"background": {
    "page": "background.html"
  }**,
  ...
}
```

如果你需要浏览器更早启动 — 例如，你想显示通知 — 那么，你也许也希望指定"background"权限。

细节

可以用类似于帧之间通讯的方式，直接使用脚本调用在一个扩展的多个页面之间进行通讯。 `chrome.extension.getViews()` 方法会返回属于你的扩展的每个活动页面的窗口对象列表，而 `chrome.extension.getBackgroundPage()` 方法返回背景页。

范例

下面的代码片段演示了扩展如何在背景页中与其他页面交互。同时也展示如何使用背景页来处理事件，如用户点击。

例子中的扩展有一个背景页，多个由image.html创建的view页面。（通过`chrome.tabs.create()`）。

```
_//In background.js:_
// React when a browser action's icon is clicked.
chrome.browserAction.onClicked.addListener(function(tab) {
  var viewTabUrl = chrome.extension.getURL('image.html');
  var imageUrl = _/* an image's URL */_;
  // Look through all the pages in this extension to find one we can use.
  var views = chrome.extension.getViews();
  for (var i = 0; i < views.length; i++) {
    var view = views[i];
    // If this view has the right URL and hasn't been used yet...
    if (view.location.href == viewTabUrl && !view.imageAlreadySet) {
      // ...call one of its functions and set a property.
      view.setImageUrl(imageUrl);
      view.imageAlreadySet = true;
      break; // we're done
    }
  }
});
_//In image.html:_
<html>
  <script>
    function setImageUrl(url) {
      document.getElementById('target').src = url;
    }
  </script>
  <body>
    <p>
    Image here:
    </p>
    <img id="target" src="white.png" width="640" height="480">
  </body>
</html>
```

Content Scripts

Contents

Content Scripts

Content scripts是在Web页面内运行的javascript脚本。通过使用标准的DOM，它们可以获取浏览器所访问页面的详细信息，并可以修改这些信息。

下面是content scipt可以做的一些事情范例：

- 从页面中找到没有写成超链接形式的url，并将它们转成超链接。
- 放大页面字体使文字更清晰
- 找到并处理DOM中的microformat

当然，content scripts也有一些限制，它们不能做的事情包括 ：

- 不能使用除了chrome.extension之外的chrome.* 的接口
- 不能访问它所在扩展中定义的函数和变量
- 不能访问web页面或其它content script中定义的函数和变量
- 不能做cross-site XMLHttpRequests

这些限制其实并不像看上去那么糟糕。Content scripts 可以使用messages机制与它所在的扩展通信，来间接使用chrome.*接口，或访问扩展数据。Content scripts还可以通过共享的DOM来与web页面通信。更多功能参见执行环境。

Manifest

如果content scipt的代码总是需要注入，可以在extension manifest中的 content_scipt字段注册它。如下面的例子：

```
{
  "name": "My extension",
  ...
  **"content_scripts": [
    {
      "matches": ["http://www.google.com/*"],
      "css": ["mystyles.css"],
      "js": ["jquery.js", "myscript.js"]
    }
  ]**,
  ...
}
```

如果只是在某些情况下需要注入，可以使用permission字段，详见Programmatic injection。

```
{
  "name": "My extension",
  ...
  **"permissions": [
    "tabs", "http://www.google.com/*"
  ]**,
  ...
}
```

使用 `content_scripts` 字段，一个扩展可以向一个页面注入多个content_script脚本；每个content script脚本可以包括多个javascript脚本和css文件。content_script字段中的每一项都可以包括下列属性：

Name	Type	Description
`matches`	array of strings	必须。 定义哪些页面需要注入content script。查看Match Patterns以了解详细语法。
`css`	array of strings	可选。需要向匹配页面中注入的CSS文件。这些文件将在页面的DOM树创建和显示之前，按照定义的顺序依次注入。
`js`	array of strings	可选。 需要向页面中注入的javascript文件，按定义顺序注入。
`run_at`	string	可选。 控制content script注入的时机。可以是document_start， document_end或者document_idle。缺省时是document_idle。如果是document_start, 文件将在所有CSS加载完毕，但是没有创建DOM并且没有运行任何脚本的时候注入。如果是document_end，则文件将在创建完DOM之后，但还没有加载类似于图片或frame等的子资源前立刻注入。如果是document_idle，浏览器会在document_end和发出window.onload事件之间的某个时机注入。具体的时机取决与文档加载的复杂度，为加快页面加载而优化。注意：在document_idle的情况下，content script不一定会接收到window.onload事件，因为它有可能在事件发出之后才加载。在大多数情况下， 在content script中监听onload事件是不必要的，因为浏览器会确保在DOM创建完成后才执行它。 如果一定要在window.onload的时候运行，可以通过document.readyState属性来检查onload事件是否已经发出。
`all_frames`	boolean	可选。控制是在匹配页面的所有frame中运行还是只在最上层的frame中运行。缺省是false， 也就是只在最上层frame中运行。
`include_globs`	array of string	可选。控制将content_script注入到哪些匹配的页面中。模拟Greasemonkey中的@include关键字。 详细信息可以查看下面的Include and exclude globs。
`exclude_globs`	array of string	可选。控制将content_script注入到哪些匹配的页面中。模拟Greasemonkey中的@exclude关键字。 详细信息可以查看下面的Include and exclude globs。

Include和exclude语句

一个content script被注入页面的条件是： 页面url匹配任意一条match模式， 并且匹配任意一条include glob模式，并且不匹配任何exclude glob模式。 由于matches属性是必选的，因此include glob和exclude glob都只能用来限制哪些匹配的页面会被影响。

另外， 这两个属性与matches属性的语法是不同的， 它们更灵活一些。 在这两个属性中可以包含号和？号作为通配符。 其中可以匹配任意长度的字符串，而？匹配任意的单个字符。

例如，语句"http://???.example.com/foo/*"" 可以匹配下面的所有情况：

- "http://www.example.com/foo/bar"
- "http://the.example.com/foo/"

但是它不能匹配下面的这些情况：

- "http://my.example.com/foo/bar"
- "http://example.com/foo/"
- "http://www.example.com/foo"

编程式注入

如果不需要将javascript 和css注入到每一个匹配的网页里面，可以通过程序来控制代码的注入。 例如， 可以只在用户点击了一个browser action图标后才注入脚本。

如果要将代码注入页面，扩展必须具有cross-origin 权限， 还必须可以使用chrome.tabs模块。 可以通过在manifest文件的permissions字段里声明来取得这些权限。

一旦设置好了权限，就可以通过调用executeScript()来注入javascript脚本。如果要注入css，可以调用insertCSS()。

下面的代码（见例子make_page_red） 演示了点击按钮后向当前标签的页面中注入并执行javascript代码。

```
_/* in background.html */_
chrome.browserAction.onClicked.addListener(function(tab) {
  chrome.tabs.executeScript(null,
                           {code:"document.body.bgColor='red'"});
});

_/* in manifest.json */_
"permissions": [
  "tabs", "http://*/*"
],
```

当浏览器显示一个http网页并且用户点击了扩展的browser action按钮后，扩展会将页面的bgcolor属性设置为红色。 如果这个页面没有用css设置它的背景颜色的话， 它会变成红色。

一般来说，可以将代码放在文件里面而不是像上面那个例子那样直接注入。 可以这样写：

```
chrome.tabs.executeScript(null, {file: "content_script.js"});
```

执行环境

Content script是在一个特殊环境中运行的，这个环境成为isolated world（隔离环境）。它们可以访问所注入页面的DOM,但是不能访问里面的任何javascript变量和函数。 对每个content script来说，就像除了它自己之外再没有其它脚本在运行。 反过来也是成立的： 页面里的

javascript也不能访问content script中的任何变量和函数。

例如，这个简单的页面：

```
hello.html
==========
<html>
  <button id="mybutton">click me</button>
  <script>
    var greeting = "hello, ";
    var button = document.getElementById("mybutton");
    button.person_name = "Bob";
    button.addEventListener("click", function() {
      alert(greeting + button.person_name + ".");
    }, false);
  </script>
</html>
```

现在，将下面这个脚本注入hello.html：

```
contentscript.js
================
var greeting = "hola, ";
var button = document.getElementById("mybutton");
button.person_name = "Roberto";
button.addEventListener("click", function() {
  alert(greeting + button.person_name + ".");
}, false);
```

然后，如果按下按钮， 可以同时看到两个问候。

隔离环境使得content script可以修改它的javascript环境而不必担心会与这个页面上的其它content script冲突。 例如，一个content script可以包含JQuery v1而页面可以包含JQuery v2， 它们之间不会产生冲突。

另一个重要的优点是隔离环境可以将页面上的脚本与扩展中的脚本完全隔离开。这使得开发者可以在content script中提供更多的功能，而不让web页面利用它们。

与嵌入的页面通信

尽管content script的执行环境与所在的页面是隔离的，但它们还是共享了页面的DOM。 如果页面需要与content script通信（或者通过content script与扩展通信）， 就必须通过这个共享的DOM。

下面这个例子是通过自定义的DOM事件和把数据放到固定的地方来实现的：

```
http://foo.com/example.html
===========================
var customEvent = document.createEvent('Event');
customEvent.initEvent('myCustomEvent', true, true);

function fireCustomEvent(data) {
  hiddenDiv = document.getElementById('myCustomEventDiv');
  hiddenDiv.innerText = data
  hiddenDiv.dispatchEvent(customEvent);
}
```

```
contentscript.js
================
var port = chrome.extension.connect();

document.getElementById('myCustomEventDiv').addEventListener('myCustomEvent', function()
  var eventData = document.getElementById('myCustomEventDiv').innerText;
  port.postMessage({message: "myCustomEvent", values: eventData});
});
```

在上面的例子中，html页面（不属于扩展）创建了一个自定义事件， 当它向DOM中的一个特定元素写入事件数据后就会激活并派发这个自定义事件。 Content script在这个特定元素上监听这个自定义事件， 从这个元素中获取数据，并向扩展进程post一个消息。 通过这种方式，页面建立了与扩展的通信链接。 这个方法也适用于反向的通信。

安全性

在写content script的时候，有两个安全问题必须注意。 首先， 要小心不要给原页面带来新的安全漏洞。 例如， 如果content script要从其它网站获取数据（比如通过背景页面做XMLHttpRequest调用）， 在将数据注入前，应该进行处理以防止cross-site scripting攻击，比如用innerText注入而不是用innerHTML注入。特别要小心的是在一个HTTPS的页面上获取HTTP的内容，因为这个内容很可能是被人用man-in-the-middle方式破坏过的。

其次，尽管在独立环境中运行content script的机制已经提供了一些保护，如果不加区分的使用web页面上的内容还是可以被恶意的web页面攻击的。

```
contentscript.js
================
var data = document.getElementById("json-data")
// WARNING! Might be evaluating an evil script!
var parsed = eval("(" + data + ")")

contentscript.js
================
var elmt_id = ...
// WARNING! elmt_id might be "); ... evil script ... //"!
window.setTimeout("animate(" + elmt_id + ")", 200);
```

建议使用安全一些的API：

```
contentscript.js
================
var data = document.getElementById("json-data")
// JSON.parse does not evaluate the attacker's scripts.
var parsed = JSON.parse(data)

contentscript.js
================
var elmt_id = ...
// The closure form of setTimeout does not evaluate scripts.
window.setTimeout(function() {
  animate(elmt_id);
}, 200);
```

引用扩展里的文件

通过chrome.extension.getURL()来获取扩展里文件的URL。可以像使用其它url一样使用这些URL，如下面的例子所示：

```
_//Code for displaying <extensiondir>/images/myimage.png:</extensiondir>_
var imgURL = **chrome.extension.getURL("images/myimage.png")**;
document.getElementById("someImage").src = imgURL;
```

例子

例子contentscript_xhr显示了一个扩展如何从content script里进行cross-site请求。可以在examples/api/messaging 里的消息通信部分找到更多例子。

可以看make_page_red 和email_this_page 这两个例子了解程序注入。

更多例子和源代码，可以查看例子

跨域 XMLHttpRequest 请求

普通网页能够使用XMLHttpRequest对象发送或者接受服务器数据, 但是它们受限于同源策略. 扩展可以不受该限制. 任何扩展只要它先获取了跨域请求许可，就可以进行跨域请求。

注意: 页面内容脚本不能直接发起跨域请求. 然而, 任何一个页面内容脚本都可以发送消息给父扩展，请求父扩展发起一次跨域请求。关于使用这一技术的例子，请参照contentscript_xhr example.

扩展所属域

每个正在运行的扩展都存在于自己独立的安全域里. 当没有获取其他权限时，扩展能够使用XMLHttpRequest获取来自安装该扩展的域的资源. 例如, 假设有一个扩展包含一个叫config.json的JSON配置文件,该文件位于config_resources目录, 那么该扩展能够使用下面这段代码获取文件内容:

```
var xhr = new XMLHttpRequest();
xhr.onreadystatechange = handleStateChange; // Implemented elsewhere.
xhr.open("GET", chrome.extension.getURL('/config_resources/config.json'), true);
xhr.send();
```

如果某个扩展希望访问自己所属域以外的资源，比如说来自http://www.google.com的资源(假设该扩展不是来自www.google.com), 浏览器不会允许这样的请求，除非该扩展获得了相应的跨域请求允许。

获取跨域请求允许

通过添加域名或者域名匹配到manifest文件的permissions段, 该扩展就拥有了访问除了自己所属域以外的其他域的访问权限.

```
{
  "name": "My extension",
  ...
  **"permissions": [
    "http://www.google.com/"
  ]**,
  ...
}
```

跨域允许设置可以使用完整域名, 例如:

- "http://www.google.com/"
- "http://www.gmail.com/"

或者使用模式匹配, 例如:

- "http://*.google.com/"
- "http://*/"

模式匹配"http://*/" 表示可以发起到所有域的HTTP请求. 注意在这里, 模式匹配有点像内容脚本匹配, 但是这里的任何域名后的路径信息都被忽略

这里还需要注意访问权限是根据访问协议(匹配模式里的http或者https或者其他协议名)及域名来授予的. 例如某个扩展希望同时基于https和http协议访问某个域或者某些域, 那么它必须分别获取基于这两种协议的访问允许(类似下面这样的声明):

```
"permissions": [
  "http://www.google.com/",
  "https://www.google.com/"
]
```

安全性考虑

每当使用通过XMLHttpRequest获取的资源时, 你编写的背景页需要注意不要成为跨域脚本的牺牲品. 特别注意避免使用像下面这样的危险API:

```
background.html
===============
var xhr = new XMLHttpRequest();
xhr.open("GET", "http://api.example.com/data.json", true);
xhr.onreadystatechange = function() {
  if (xhr.readyState == 4) {
    // 警告! 这里有可能执行了一段恶意脚本!
    var resp = eval("(" + xhr.responseText + ")");
    ...
  }
}
xhr.send();

background.html
===============
var xhr = new XMLHttpRequest();
xhr.open("GET", "http://api.example.com/data.json", true);
xhr.onreadystatechange = function() {
  if (xhr.readyState == 4) {
    // 警告! 这样处理有可能被注入恶意脚本!
    document.getElementById("resp").innerHTML = xhr.responseText;
    ...
  }
}
xhr.send();
```

实际上我们应该首选不会执行脚本的安全API:

```
background.html
===============
var xhr = new XMLHttpRequest();
xhr.open("GET", "http://api.example.com/data.json", true);
xhr.onreadystatechange = function() {
  if (xhr.readyState == 4) {
    // JSON解析器不会执行攻击者设计的脚本.
    var resp = JSON.parse(xhr.responseText);
  }
}
xhr.send();

background.html
===============
var xhr = new XMLHttpRequest();
xhr.open("GET", "http://api.example.com/data.json", true);
xhr.onreadystatechange = function() {
  if (xhr.readyState == 4) {
    // innerText不会给攻击者注入HTML元素的机会.
    document.getElementById("resp").innerText = xhr.responseText;
  }
}
xhr.send();
```

另外在使用通过协议HTTP获取的资源时要特别小心. 如果你开发的扩展被应用在恶意网络环境中，网络攻击者(又叫 "中间人攻击") 可能篡改服务器响应内容从而可能攻击你编写的扩展. 事实上，你应该尽可能地首选使用HTTPS协议.

国际化 (i18n)

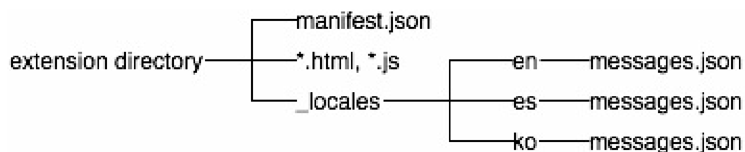
目录

国际化 (i18n)

一个支持国际化的扩展能很容易被本地化，但原始不支持自动确认语言和地区。

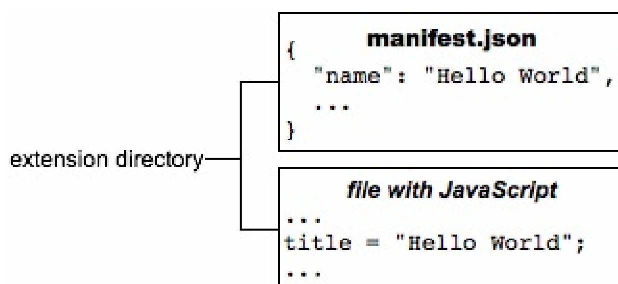
为了国际化您的扩展,您需要把所有用户可见字符串保存在文件名为messages.json的文件里。每当你本地化您的扩展时，您需要在*locales/localeCode*下增加这个*messages.json*文件，*lo_caleCode* 是一个形如en代表英语的编码。

下面是一个支持英语(en)、西班牙语(es)和韩语(ko)的国际化扩展文件层次结构图。



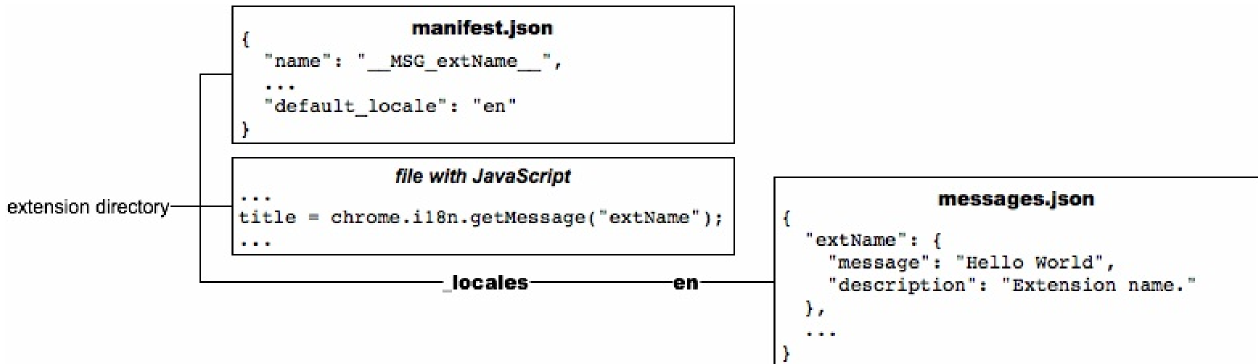
如何支持多种语言

假设您有一个如下图所列文件的扩展:



为了国际化这个扩展，您需要先命名每个用户可见的字符串，然后将它们保存在messages.json文件里。这个扩展的manifest、CSS和JavaScript代码文件就能根据每个字符串的名字提取到他们的本地化版本。

下图是一个国际化扩展的文件规范示意图（它只支持英语字符串）



重要提示:如果一个扩展有_locales目录，那么manifest文件必须定义"default_locale"字段内容。

国际化扩展的一些方法、规则或技巧:

- 你能使用任何一种被支持的 locales. 如果您使用未支持的locale，Google Chrome会忽略它。
- 在manifest.json和CSS文件中，像下图一样引用一个字符串：

```
__MSG__messagename___
```

- 在您的扩展JavaScript程序中，像下图一样引用一个字符串：

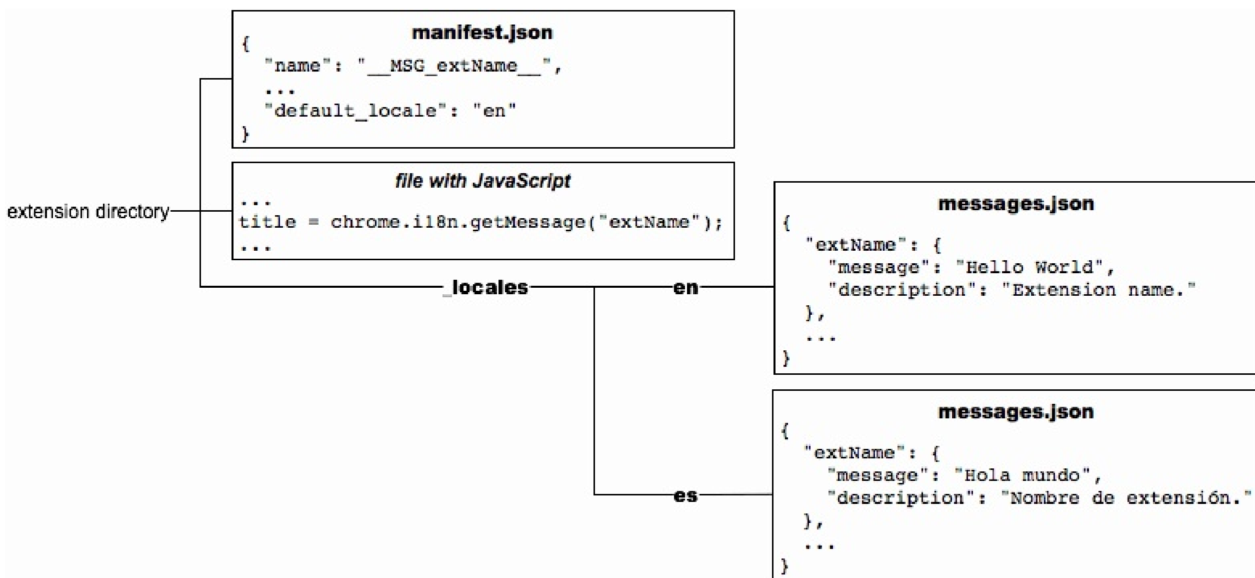
```
chrome.i18n.getMessage("_messagename_")
```

- 每次调用getMessage()，最多能返回9个字符串。查看Examples: getMessage连接了解更多信息
- 国际化系统已经提供了一些 message, 像 @@bidi_dir 和 @@ui_locale。 查看 Predefined messages 连接了解全部预定义message 名称。
- 在message.json文件里，每个用户可用的字符串都有一个名字项，一个"message"项和一个可选的"description"项。这个名字如"extName"或"search_string"是字符串的ID, "message"是字符串在当前语言环境中的值。"description"有一些解释信息。示例：

```
{
  "search_string": {
    "message": "hello%20world",
    "description": "The string we search for. Put %20 between words that go together.
  },
  ...
}
```

了解更多信息，请查看Formats: Locale-Specific Messages.

一旦国际化了扩展，就很容易支持其它语言了。1、拷贝一个message.json,翻译其字符串，然后将它保存在_locales目录下。例如:为了支持西班牙语，只要翻译好一个message.json，放在_locales/es目录下。下图展示了上述扩展在支持了西班牙语以后的文件层次图。



预定义 messages

国际化系统提供一些预定义messages来帮助您本地化扩展。这些包括@@ui*locale(*您能用它检测当前*UI*系统的语言信息*),*也包括一些以@@*bidi*作前部的message(用来检测文字方向书写习惯，像英语的是从左到右书写习惯)，它们(@@bidi_)有相似的名字， 例举于:gadgets BIDI (bi-directional) API.

@@extension_id message 能用在任意扩展 的CSS和JavaScript 文件中，无论其是否已经本地化，但不能用在manifest.json文件中。

注意:Content script CSS files 不能用一些预定义message，像 @@extension_id。更多信息参见bug 39899.

下表描述了每个预定义的message。

Message 名字	描述
@@extension_id	这个扩展的 ID; 您可以在扩展内部用这个字符串构造URL用于访问某些资源。 没有本地化的扩展也可以使用这个message.

注意: 您不能在manifest文件中使用这个message | | @@ui_locale | 当前语言; 您可以用这个字符串构造和语言环境相关的URL。 | | @@bidi_dir | 当前语言环境的文字方向, 要么是 "ltr" (代表从左到向的语言，如英语)。要么是rtl" (代表右到左的语言，如日语)。 | | @@bidi_reversed_dir | 其值是对@@bidi_dir取反。如果 @@bidi_dir 是 "ltr", 那么它是 "rtl", 否则它是 "ltr".。 | | @@bidi_start_edge | 如果 @@bidi_dir 是 "ltr", 那么它是 "left"， 否则它是 "right". | | @@bidi_end_edge | 如果 @@bidi_dir 是 "ltr", 那么它是 "right";， 否则它是 "left". |

一个在CSS文件中，用@@extension_id 构造一个URL的示例：

```
body {
  **background-image:url('chrome-extension://__MSG_@@extension_id__/background.png');**
}
```

如果扩展的ID是abcdefghijklmnopqrstuvwxyzabcdef，那么上述代码中的粗线部份变成：

```
background-image:url('chrome-extension://abcdefghijklmnopqrstuvwxyzabcdef/background.png'
```

一个在CSS文件中用@@bidi_* messages 的例子：

```
body {
  **direction: __MSG_@@bidi_dir__;**
}

div#header {
  margin-bottom: 1.05em;
  overflow: hidden;
  padding-bottom: 1.5em;
  **padding-__MSG_@@bidi_start_edge__: 0;**
  **padding-__MSG_@@bidi_end_edge__: 1.5em;**
  position: relative;
}
```

从左到右的语言，如英语。粗线变成：

```
dir: ltr;
padding-left: 0;
padding-right: 1.5em;
```

语言环境

扩展能使用所有Google Chrome支持的语言，,附加几个有几种地区变化的语言(如en : en_GB代表英式英语， en_US代表美式英语)。

已支持的语言

你的扩展能使用下列的任一种语言:

am ar bg bn ca cs da de el en en_GB en_US es es_419 et fi fil fr gu he hi hr hu id it ja kn ko ltlv ml mr nl or pl pt pt_BR pt_PT ro ru sk sl sr sv sw ta te th tr uk vi zh zh_CN zh_TW

扩展怎么查找字符串

您不是必须为每个语言定义各个字符串。支持国际化的扩展自带自动搜索功能，只要默认语言的message.json文件中字义了这些字符串，无论是什么语言，您的扩展都能运行起来。下面介绍扩展系统怎么搜索message.json：

1. 在用户首选的存在的语言中搜索messages文件 。举个例子：如果Google Chrome的语言设置为英式英语(en_GB),系统会先查找_locales/en_GB/messages.json，如果这个文件和其message存在，系统停止搜索。
2. 如果用户首选的语言编码有一个带下划线的地区标识码。如:en_US。那么系统会再次搜索en(不带US)的message文件。果这个文件和message存在，系统停止搜索。
3. 搜索扩展默认语言的messages 文件。举例：默认语言是 "ES"，那么，搜索 _locales/es/messages.json.。

下图中，"colores"的message有三种语言支持，"extName"有两种。美式英语用户始终会看到标签"Colors"，英式英语用户会看到标签"Clours"，他们会同时看到标签"Hello World"。因为默认语言是西班牙语，所以非英语用户会看到标签"Colores"和扩展名称"Hola mundo"。

manifest.json

```
{
  ...
  "default_locale": "es"
}
```

_locales/es/messages.json

```
{
  "extName": {
    "message": "Hola mundo",
    "description": "Nombre de extensión."
  },
  "colores": {
    "message": "Colores",
    "description": "Etiqueta."
  },
  ...
}
```

_locales/en/messages.json

```
{
  "extName": {
    "message": "Hello World",
    "description": "Extension name."
  },
  "colores": {
    "message": "Colors",
    "description": "Label."
  },
  ...
}
```

_locales/en_GB/messages.json

```
{
  "colores": {
    "message": "Colours",
    "description": "Label."
  }
}
```

怎么设置浏览器的语言

为了测试翻译，您可能需要设置浏览器的语言.这部份介绍怎么在Windows,Mac OS X, and Linux上设置语言。

Windows

你可以在locale-specific快捷方式或Google Chrome UI上改变语言。用快捷方式建立更快，并且这种方法能同时使用多种语言。

使用**locale-specific**快捷方式

创建和使用快捷方式，运行特定语言的Google Chrome：

1. 复制一个Google Chrome的桌面快捷方式。
2. 用语言编码命名新的快捷方式，如cn_chrome.lnk。
3. 选择快捷方式的连接属性，加入--lang和--user-data-dir参数。如：

```
_path_to_chrome.exe_ --lang=_locale_ --user-data-dir=c:_locale_profile_dir_
```

4. 双击新的快捷方式运行Chrome。

示例：用西班牙语创建此类快捷方式:

```
_path_to_chrome.exe_ --lang=es --user-data-dir=c:chrome-profile-es
```

示例：你可以创建任意多的快捷方式,，让你的多种语言测试变很容易。

```
_path_to_chrome.exe_ --lang=en --user-data-dir=c:chrome-profile-en
_path_to_chrome.exe_ --lang=en_GB --user-data-dir=c:chrome-profile-en_GB
_path_to_chrome.exe_ --lang=ko --user-data-dir=c:chrome-profile-ko
```

注意:不是必须指定--user-data-dir，但是正如您所见，加这个参数也很容易。并且每为个语言指定一个user-data-dir目录，可以让你在同一时间运行几种语言的浏览器。 不过有一个缺点，因为语言数据是不共享的，所以您需要安装多次扩展(需要为为每种语言安装一次)。呵呵，当然你不想使用这种语言也可以不安装。查看Creating and Using Profiles连接获得更多信息。

使用 **UI**

怎么使用windows版本Google Chrome UI 改变语言

1. 设置按钮（扳手图标）->选项
2. 选择当前标签页
3. 将页面滚动到底
4. 点击"改变字符和语言设置"
5. 选择语言标签
6. 使用下拉列表设置Google Chrome语言
7. 重启Chrome

Mac OS X

在Mac上改变语言，您需要使用系统preferences

1. 从Apple menu上选择**System Preferences**
2. 在**Personal**部份,选择**International**
3. 选择你的语言和地域
4. 重启chrome

Linux

Linux上改变语言，先退出Google Chrome，再如下设置环境变量，重启即可:

```
LANGUAGE=es ./chrome
```

一些例子

您能轻松的在examples/api/i18n目录下找到国际化的例子。xamples/extensions/news目录下有更完整的例子。samples有代码级的其它例子和帮助。

函数 getMessage示例

下面的代码介绍了怎么得到和显示一个本地化的字符串。用"string1"和"string2"替换message的两个占位符。

```
function getMessage() {
  var message = chrome.i18n.getMessage("click_here", ["string1", "string2"]);
  document.getElementById("languageSpan").innerHTML = message;
}
```

如何提供和使用一个字符串。

```
_// In JavaScript code_
status.innerText = chrome.i18n.getMessage("error", errorDetails);

_// In messages.json_
"error": {
  "message": "Error: $details$",
  "description": "Generic error template. Expects error parameter to be passed in.",
  "placeholders": {
    "details": {
      "content": "$1",
      "example": "Failed to fetch RSS feed."
    }
  }
}
```

Locale-Specific Messages 了解更多占位符的信息。关于调用函数getMessage()的更多信息参见API reference.

函数getAcceptLanguages示例

下例代码介绍了怎么取得所有可用的语言，把它们以","号连接成字符串并显示。

```
function getAcceptLanguages() {
  chrome.i18n.getAcceptLanguages(function(languageList) {
    var languages = languageList.join(",");
    document.getElementById("languageSpan").innerHTML = languages;
  })
}
```

调用函数getAcceptLanguages()的更多细节请参见API reference.

API 介绍: chrome.i18n

方法

getAcceptLanguages

chrome.i18n.getAcceptLanguages(functioncallback)

取得所有的可用语言，这个和浏览器使用的语言不同， 用window.navigator.language.取得这浏览器使用的语言

参数

callback*(function)*

callback function

这个回调参数应该是一个函数，类似于：

```
function(array of string languages) {...};
```

languages*(array of string)*

浏览器可接受的语言数组, 类似于 en-US,en,zh-CN

getMessage

string chrome.i18n.getMessage(string messageName, string or array of string substitutions)

用指定的message取得字符串。如果message不存在，此方法返回空串。如果调用格式错误，如*messageName*不是一个串或是*substitution*为空或大于9个元素。此函数返回"undefined"

参数

messageName*(string)*

message名字, 在messages.json文件中使用的名字

substitutions*(optionalstring or array of string)*

如果message需要这个参数，设定1到9个子串。

返回

(string)

当前语言设置的Message

消息传递

目录

For information on how to use experimental APIs, see the chrome.experimental.* APIs page.

消息传递自从content script内容运行在网页环境而不是在扩展中，我们经常需要一些方法和其余的扩展进行通信。例如，一个 RSS阅读扩展可能会使用content scripts去检测RSS阅读扩展应该提供给哪个页面，然后通知后台页面以便显示一个页面交互图标。

通过使用消息传递机制在扩展和content scripts中通信，任何一方可以收到另一方传递来的消息，并且在相同的通道上答复。这个消息可以包含 任何一个有效的JSON对象(null, boolean, number, string, array, or object)。这里有一些简单的API对于 一次简单的请求，还有更多复杂的API，它们可以帮助你长时间保持连接在一个共享的环境中交换大量的信息，它也可以帮助你传递消息给另外一个你知道ID的扩展，在扩展之间的消息传递这里会有详细的讲解。

一次简单的请求

如果你仅仅需要给你自己的扩展的另外一部分发送一个消息（可选的是否得到答复），你可以简单地使用chrome.extension.sendRequest()或者chrome.tabs.sendRequest()方法。这个方法可以帮助你传送一次JSON序列化消息从content script到扩展，反之亦然。如果接受消息的一方存在的话，可选的回调参数允许处理传回来的消息。

像下面这个例子一样，可以从content script 发起一个请求：

```
contentscript.js
================
chrome.extension.sendRequest({greeting: "hello"}, function(response) {
  console.log(response.farewell);
});
```

传递一个请求到扩展很容易，你需要指定哪个标签发起这个请求。下面这个例子展示了如何指定标签发起一个请求。

```
background.html
===============
chrome.tabs.getSelected(null, function(tab) {
  chrome.tabs.sendRequest(tab.id, {greeting: "hello"}, function(response) {
    console.log(response.farewell);
  });
});
```

接受消息的一方，需要启动一个chrome.extension.onRequest事件监听器用来处理消息。这个方法在content script和扩展中都是一样的。这个请求将会保留直到你做出了回应。下面的这个例子是一个很好的做法调用一个空对象请求然后得到答复的例子。

```
chrome.extension.onRequest.addListener(
  function(request, sender, sendResponse) {
    console.log(sender.tab ?
                "from a content script:" + sender.tab.url :
                "from the extension");
    if (request.greeting == "hello")
      sendResponse({farewell: "goodbye"});
    else
      sendResponse({}); // snub them.
  });
```

小提示：如果多个页面都发起了相同的请求，都在等待答复，只有第一个发起请求的页面会得到响应，其他的将会被忽略。

长时间的保持连接

有时候持续长时间的保持会话会比一次简单的请求有用。你可以建立一个长时间存在的通道从content script到扩展, 反之亦然,使用chrome.extension.connect()或者chrome.tabs.connect()方法，你可以把这个通道命名，为了更方便区分不同类型的连接。

一个有用的例子就是自动填写表单扩展。content script可以建立一个通道在登录页面和扩展之间，同时发出一条消息给扩展，告诉扩展需要填写的内容。共享的连接允许扩展保持共享状态，从而连接几个来自content script.的消息。

当建立连接，两端都有一个Port 对象通过这个连接发送和接收消息。

下面展示了如何从content script建立一个通道，发送和接受消息：

```
contentscript.js
================
var port = chrome.extension.connect({name: "knockknock"});
port.postMessage({joke: "Knock knock"});
port.onMessage.addListener(function(msg) {
  if (msg.question == "Who's there?")
    port.postMessage({answer: "Madame"});
  else if (msg.question == "Madame who?")
    port.postMessage({answer: "Madame... Bovary"});
});
```

从扩展到content script发送一个请求看起来非常简单，除了你需要指定哪个标签需要连接，简单的办法就是上面例子中的chrome.tabs.connect(tabId, {name: "knockknock"}).

为了处理正在等待的连接，你需要用chrome.extension.onConnect 事件监听器，对于content script或者扩展页面，这个方法都是一样的，但你的扩展的另外一个部分调用"connect()"， 这个事件一旦被触发，通过这个连接你可以利用Port对象进行发送和接收消息，下面的例子展示了如何处理连接：

```
chrome.extension.onConnect.addListener(function(port) {
  console.assert(port.name == "knockknock");
  port.onMessage.addListener(function(msg) {
    if (msg.joke == "Knock knock")
      port.postMessage({question: "Who's there?"});
    else if (msg.answer == "Madame")
      port.postMessage({question: "Madame who?"});
    else if (msg.answer == "Madame... Bovary")
      port.postMessage({question: "I don't get it."});
  });
});
```

你可能想知道这个连接什么时候被关闭。例如，如果你在为每个开放的端口维护分开的状态，如果你想知道它什么时候关闭，因此，需要监听Port.onDisconnect 事件，这个事件被激发，要么是通道调用了Port.disconnect()这个方法，要么这个页面包含的端口没有被加载(例如这个标签是个导航栏)，onDisconnect()保证仅仅被激发一次。

扩展之间的消息传递

除了在扩展的组件之间传送消息，你还可以使用消息API来和其他的扩展之间进行通信，你可以使用一个其他扩展也可以利用的公共API。

对于扩展内部来说，监听一个传入的请求和连接是一样的，你可以使用chrome.extension.onRequestExternal或者chrome.extension.onConnectExternal方法，如下面的例子所示：

```
// For simple requests:
chrome.extension.onRequestExternal.addListener(
  function(request, sender, sendResponse) {
    if (sender.id == blacklistedExtension)
      sendResponse({});  // don't allow this extension access
    else if (request.getTargetData)
      sendResponse({targetData: targetData});
    else if (request.activateLasers) {
      var success = activateLasers();
      sendResponse({activateLasers: success});
    }
  });

// For long-lived connections:
chrome.extension.onConnectExternal.addListener(function(port) {
  port.onMessage.addListener(function(msg) {
    // See other examples for sample onMessage handlers.
  });
});
```

同样，传递一个消息到另外一个扩展和把消息传递给自己扩展的另外一部分是一样的，唯一不同的是你必须知道你要传给消息的扩展的ID例如：

```
// The ID of the extension we want to talk to.
var laserExtensionId = "abcdefghijklmnoabcdefhijklmnoabc";

// Make a simple request:
chrome.extension.sendRequest(laserExtensionId, {getTargetData: true},
  function(response) {
    if (targetInRange(response.targetData))
      chrome.extension.sendRequest(laserExtensionId, {activateLasers: true});
  });

// Start a long-running conversation:
var port = chrome.extension.connect(laserExtensionId);
port.postMessage(...);
```

安全策略

无论是从content script还是从扩展接收消息，你的页面不应该cross-site scripting，特别是避免使用那些不安全的API例如下面的例子：

```
background.html
===============
chrome.tabs.sendRequest(tab.id, {greeting: "hello"}, function(response) {
  // WARNING! Might be evaluating an evil script!
  var resp = eval("(" + response.farewell + ")");
});

background.html
===============
chrome.tabs.sendRequest(tab.id, {greeting: "hello"}, function(response) {
  // WARNING! Might be injecting a malicious script!
  document.getElementById("resp").innerHTML = response.farewell;
});
```

相反的，选择更安全的API而不是运行脚本。

```
background.html
===============
chrome.tabs.sendRequest(tab.id, {greeting: "hello"}, function(response) {
  // JSON.parse does not evaluate the attacker's scripts.
  var resp = JSON.parse(response.farewell);
});

background.html
===============
chrome.tabs.sendRequest(tab.id, {greeting: "hello"}, function(response) {
  // innerText does not let the attacker inject HTML elements.
  document.getElementById("resp").innerText = response.farewell;
});
```

范例

你可以在这个examples/api/messaging 目录下找到消息传递的例子，也可以找到contentscript_xhr (contents script和扩展之间传递消息的例子)，更多的内容和源码，请查看Smples.

API reference: chrome.apiname

Properties

getLastError

chrome.extensionlastError

Methods

method name

void chrome.module.methodName(, ``)

Undocumented.

A description from the json schema def of the function goes here.

Parameters

Returns

Callback function

The callback *parameter* should specify a function that looks like this:

If you specify the *callback* parameter, it should specify a function that looks like this:

```
function(Type param1, Type param2) {...};
```

This function was added in version . If you require this function, the manifest key minimum_chrome_version can ensure that your extension won't be run in an earlier browser version.

Events

event name

chrome.bookmarksonEvent.addListener(function(Type param1, Type param2) {...});

Undocumented.

A description from the json schema def of the event goes here.

Parameters

Types

type name

Optional Permissions

内容

`Chrome.permissions` 用于实现可选权限。在您扩展的运行过程中，而不是在安装的时候请求权限。这能帮助用户清楚为什需要此权限，且仅在必要的时候才运行扩展使用此权限。

关于权限的相关信息及每个权限的详细信息，可参见manifest章节的permissions。

实现可选权限

步骤1：确定哪些权限作为可选，哪些权限作为必选。

为了满足基本的功能，扩展需要一些必须的权限，而另一些权限则可以在扩展运行过程再请求用户授予。

可选权限的优势：

- 扩展激活时仅仅需要少量的权限，扩展运行中需要时才请求用户授予更多的权限。
- 当扩展运行过程中请求更多权限时，可以更清楚地向用户解释为什么需要这些特定的权限。

- 可以避免Chrome浏览器禁止扩展升级（原因：当一个扩展的新版本相比老版本需要更多必选权限时，Chrome会阻止这样的扩展的自动升级）。

必选权限的优点：

- 扩展可以一次性提示用户接受所需的权限。
- 扩展运行过程可以确保拥有相关权限，从而简化了扩展的开发。

步骤2：在manifest文件中声明可选权限

在extension manifest中用 `optional_permissions` 关键字声明可选权限，与声明permissions相同：

```
{
        "name": "My extension",
        ...
        **"optional_permissions": [ "tabs", "http://www.google.com/" ],**
        ...
}
```


您可以指定下列任何可选权限：

- *host permissions*
- appNotifications
- background
- bookmarks
- clipboardRead
- clipboardWrite
- contentSettings
- contextMenus
- cookies
- debugger
- history
- idle
- management
- notifications
- pageCapture
- tabs
- topSites
- webNavigation
- webRequest
- webRequestBlocking

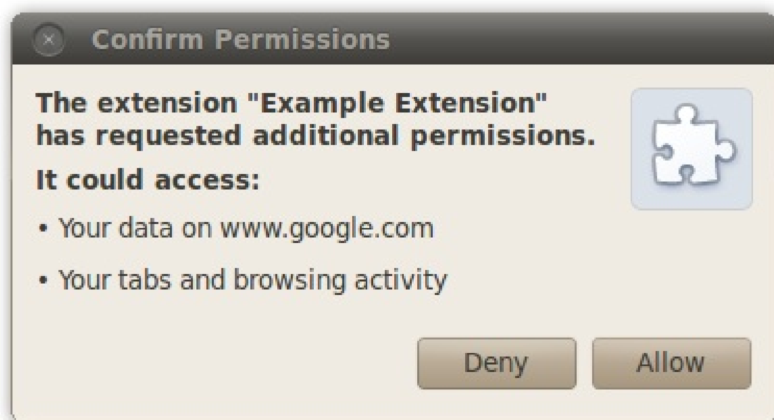
版本说明：此列表适用于Chrome 17。在之后的版本中会提供更多的可选权限。

步骤3：请求可选权限

通过调用 `permissions.request()` 请求权限，并且需要获得用户授权：

```
document.querySelector('#my-button').addEventListener('click', function(event) {
        // Permissions must be requested from inside a user gesture, like a button's
        // click handler.
        Chrome.permissions.request({
          permissions: ['tabs'],
          origins: ['http://www.google.com/']
        }, function(granted) {
          // The callback argument will be true if the user granted the permissions.
          if (granted) {
            doSomething();
          } else {
            doSomethingElse();
          }
        });
});
```


如果添加与用户看到和接受的权限不同。Chrome会提示用户warning messages。比如，上面的示例代码会导致这样的提示：



步骤4：检查扩展的当前权限

检查扩展是否拥有特定的权限，可以通过 `permission.contains()` 实现：

```
Chrome.permissions.contains({
        permissions: ['tabs'],
        origins: ['http://www.google.com/']
      }, function(result) {
        if (result) {
          // The extension has the permissions.
        } else {
          // The extension doesn't have the permissions.
        }
});
```


步骤5：删除权限

您应该删除不再需要的权限。当某个用户已授权权限被删除后，使用 `permissions.request()` 再次添加此权限时不会再提示用户。

```
Chrome.permissions.remove({
        permissions: ['tabs'],
        origins: ['http://www.google.com/']
      }, function(removed) {
        if (removed) {
          // The permissions have been removed.
        } else {
          // The permissions have not been removed (e.g., you tried to remove
          // required permissions).
        }
});
```


API 参考：Chrome.permissions

方法

contains

Chrome.permissions.contains(Permissions `permissions` , function `callback`)

检查是否拥有某些权限。

参数

`permissions` (*Permissions*)

Undocumented.

`callback` (*function*)

Undocumented.

Callback function

如果需要指定回调函数，则回调函数格式如下：

```
function(boolean result) {...};
```

`result` (*boolean*)

如果扩展已拥有指定的权限，返回true。

getAll

Chrome.permissions.getAll(function `callback`)

获取扩展当前的权限。

参数

`callback` *(function)*

Undocumented.

Callback function

如果需要指定回调函数，则回调函数格式如下：

```
function(Permissions permissions) {...};
```

`permissions` *(Permissions)*

扩展当前拥有的权限。

remove

Chrome.permissions.remove(Permissions `permissions` , function `callback`)

删除指定的权限。如果在删除过程中出现异常，Chrome.extension.lastError将会被设置

参数

`permissions` *(Permissions)*

Undocumented.

`callback` *(optional function)*

Undocumented.

Callback function

如果需要指定回调函数，则回调函数格式如下：

```
function(boolean removed) {...};
```

`removed` *(boolean)*

如果成功删除，返回true。

request

Chrome.permissions.request(Permissions `permissions` , function `callback`)

请求指定的权限。所请求的权限必需包含在manifest文件的optional_permissions里。如果在删除过程中出现异常，Chrome.extension.lastError将会被设置

参数

`permissions` (*Permissions*)

Undocumented.

`callback` (*optional function*)

Undocumented.

Callback function

如果需要指定回调函数，则回调函数格式如下：

```
function(boolean granted) {...};
```

`granted` (*boolean*)

如果用户授予请求的权限，返回true。

事件

onAdded

Chrome.permissions.onAdded.addListener(function(Permissions permissions) {...});

在扩展获得新的权限时触发。

参数

`permissions` (*Permissions*)

新获得的权限。

onRemoved

Chrome.permissions.onRemoved.addListener(function(Permissions permissions) {...});

Fired when access to permissions has been removed from the extension.

参数

`permissions` (*Permissions*)

被删除的权限。

类型

Permissions

(*object*)

Undocumented.

`permissions` (*optional array of string*)

权限名列表（不包括hosts或者origins）。

`origins` (*optional array of string*)

原始权限列表。

NPAPI 插件

使用HTML和JavaScript开发新扩展是十分容易的事情，不过如果你想在扩展中重用已经开发完成的代码和功能，你可以通过使用NPAPI插件到达目的。NPAPI插件使JavaScript代码能够调用本地二进制代码。

警告

NPAPI 是重型武器，当别的方法无法到达你的目的时，才建议使用。

运行在NPAPI插件中的代码拥有当前用户的全部权限，不能利用Google Chrome 的沙箱技术和其他安全防护技术。 在处理不可信任的输入， 如content scripts和XMLHttpRequest 时，你必须格外小心。

鉴于使用NPAPI可能引入的风险，使用了NPAPI的扩展在提交给web store或者extension gallery 时要经过人工审核。

更多信息

如何开发一个NPAPI插件超出了本文的范畴，具体请参看：Mozilla's NPAPI plugin reference 以获得更多帮助。

如果你已经拥有一个NPAPI插件，通过如下步骤，你的扩展将能够调用它。

1. 在你扩展的manifest.json文件中加入一个节，描述如何找到你的插件，以及其他一些信息，：

   ```
   {
     "name": "My extension",
     ...
     **"plugins": [
       { "path": "content_plugin.dll", "public": true },
       { "path": "extension_plugin.dll" }
     ]**,
     ...
   }
   ```

   "path" 属性用于描述如何找到你的插件，路径是相对于manifest文件位置的。 "public" 属性指明是否允许普通页面加载你的插件，默认是false，也就是只有你的扩展才能加载这个插件。

2. 创建一个HTML文件，mime-type为：application/x-my-extension"， 用于加载你的插件。

```
&lt;embed type="application/x-my-extension" id="pluginId"&gt;
&lt;script&gt;
  var plugin = document.getElementById("pluginId");
  var result = plugin.myPluginMethod();  // call a method in your plugin
  console.log("my plugin returned: " + result);
&lt;/script&gt;
```

这个页面可以被后台页面包含，或者任何你扩展会用到的其他页面。如果你的插件的 "public" 属性是true，，你可以通过脚本在页面中直接使用它。

安全注意事项

在你的扩展中包含一个NPAPI插件是一件危险的事情。因为NPAPI插件拥有访问你本地机器的完全权限而不受控制。如果你的插件不够健壮，包含漏洞，黑客可以通过溢出攻击利用漏洞来安装恶意软件到用户的机器。 有鉴于此，请尽可能的避免在扩展中使用NPAPI插件。

将NPAPI的 "public" 属性设置为true，也会增加你扩展受到攻击的可能性。因为这样一来，插件直接暴露给了页面内容，恶意网站能通过页面直接操纵你的插件。有鉴于此，尽量避免将"public"属性设置为true。

完成并发布应用

自动升级

Contents

我们希望扩展能自动升级，理由和让chrome自动升级一样：修改程序bug和安全漏洞， 增加新功能，提升性能，改善体验。

如果你通过Chrome Developer Dashboard,发布你的扩展，可以忽略此页。你可以使用Dashboard发布更新过的版本给用户，就像在Extensions Gallery 和Chrome Web Store面一样。.

概述

- 一个扩展的manifest文件里面必须指定一个"update_url"来执行升级检测。
- 扩展可以托管在Chrome Web Store，也可以发布到极速浏览器应用开放平台上。
- 如果托管在Chrome Web Store则update_url应该是：http://clients2.google.com/service/update2/crx
- 如果托管在极速浏览器应用开放平台上则update_url应该是：http://upext.chrome.360.cn/intf.php?method=ExtUpdate.query

为了能够快速和稳定的更新升级，我们建议将扩展托管在360服务器。

注：如果不指定**update_url**，在使用**360**同步服务的情况下安装扩展，重启浏览器后导致扩展丢失。

Update URL 升级url

如果将扩展托管在360服务器上，那么你需要在manifest.json文件里面加一个update_url字段，如下：

```
{
    "name": "My extension",
    ...
  **"update_url": "http://upext.chrome.360.cn/intf.php?method=ExtUpdate.query"**,
    ...
}
```

升级 manifest

服务器返回的升级manifest xml格式文件看起来如下：

```
<?xml version='1.0' encoding='UTF-8'?>
<gupdate xmlns='http://www.google.com/update2/response' protocol='2.0'>
  <app appid='**aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa**'>
    <updatecheck codebase='**http://myhost.com/mytestextension/mte_v2.crx**' version='**2
  </app>
</gupdate>
```

(这个xml格式是从google升级系统Omaha那边借用过来的，具体见 http://code.google.com/p/omaha/)

appid 属性appid是一这个扩展的id，基于扩展的公钥hash计算而来，在Packaging。里面介绍过，你可以找到你的扩展的id，通过**Chrome://extensions**.ackaing里面介绍过。

codebase 属性codebase是crx文件的url。

version 这个属性被客户端用来判断是否真的要从codebase下载crx文件。这个值必须和crx文件的 manifest.json里面的版本参数吻合。

一个升级manifest xml文件可以包含多个扩展的信息，通过多个app标记。

测试

缺省的升级检测频率是每小时一次。你可以通过扩展页面的现在立刻升级扩展来强制升级。

另外一种选择是通过命令行参数--extensions-update-frequency来设置更加频繁的升级间隔，单位秒。例如，每45秒检测一次，你可以用这样的命令行参数来运行chrome：

```
chrome.exe **--extensions-update-frequency=45**
```

注意这个将影响所有的已经安装的扩展，因此请斟酌这样做带来的带宽和服务器负载的影响。你可能想临时卸载你所有的扩展除了正在调试的，在正常浏览器使用中不应该用这个选项来运行。

高级使用：请求参数

最基本的升级机制在服务端被设计的极其简单，就是往web服务器-如apache上放就是一个静态xml文件，然后升级这个xml文件如果你的扩展有新版本了。

高级开发者可能希望充分利用升级机制，通过往升级manifest url请求参数里面添加参数来识别扩展的id和版本，他们能使用运行在一个动态页面来代替静态xml以使用同一个url来控制所有的扩展的升级。

请求参数格式：

?x=<extension_data>

其中extension_data是一个编码过的url字符串，其格式：

id=<id>&v=</id>

对于这个例子，说明我们有两个已经安装的扩展

Extension 1 with id "aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa" and version "1.1" Extension 2 with id "bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb" and version "0.4"

两个拓展都使用同一个update_url:http://test.com/extension_updates.php

两个请求看起来像这样：

http://test.com/extension_updates.php?x=id%3Daaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa%26v%3D1.1
http://test.com/extension_updates.php?x=id%3Dbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb%26v%3D0.4

在3.0.196.x之前的版本中有一个bug，地方在合并多个参数上。 (http://crbug.com/17469).

将来的工作

虽然没有实现，我们还是在一个独立的请求列出多个扩展。对于上面的例子，这个请求看起来像这样：

http://test.com/extension_updates.php?x=id%3Daaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa%26v%3D1.1&x=id%3Dbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb%26v%3D0.4

如果使用同一个update_url的扩展太多，导致请求的URL太长（超过1024字符），升级检测将发起POST请求，同时把请求参数打包在POST里面。

高级使用:最小浏览器版本

随着我们添加越来越多的api到扩展系统，你可能希望发布一个升级过的扩展，仅仅只允许运行在新版本的浏览器上。虽然chrome是自动升级的，但所有用户升级到新版浏览器也需要很多天时间。为了确保一个指定的扩展升级只作用在特定的比较高的版本上，你可以添加**prodversionmin**到你的升级manifest文件里面的app标签。例如：

```
<?xml version='1.0' encoding='UTF-8'?>
<gupdate xmlns='http://www.google.com/update2/response' protocol='2.0'>
  <app appid='aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa'>
    <updatecheck codebase='http://myhost.com/mytestextension/mte_v2.crx' version='2.0' **
  </app>
</gupdate>
```

这个配置确保只有运行在3.0.193.0以上的用户才升级到2.0版本的扩展。

托管

本页告诉你如何在自己的服务器上托管 .crx文件。如果你仅仅通过Chrome Web Store发布扩展，应用，或者主题那么你不需要本页。取而代之的是查阅Chrome Web Store帮助和开发者文档。

注意:如果你已经把扩展发布到扩展库,扩展就会合并到Chrome Web Store里。

按照惯例, 无论是Chrome Web Store还是特定服务器所提供的扩展, 可安装的web apps, 以及主题都是.crx文件。 当你使用Chrome开发者面板上传ZIP文件的时候，面板会创建.crx文件。

如果你不是使用面板来发布，那么你需要像打包中所描述的那样自己创建.crx文件。你也可以指定自动更新信息，确保你的用户可以得到最新的.crx文件副本。

一个服务器托管.crx文件必须使用适当的HTTP头,这样用户能够通过点击一个连接进行安装。

如果下列任何一种情况成立，Google Chrome认为一个文件是可安装的：

- 文件有application/x-chrome-extension内容类型
- 文件后缀是.crx并且下列两个条件都成立：
  - 文件未被送达的HTTP头X-Content-Type-Options: nosniff
  - 文件被送达的内容类型是下列之一:
    - empty string
    - "text/plain"
    - "application/octet-stream"
    - "unknown/unknown"
    - "application/unknown"
    - "*/*"

最常见的识别一个可安装的文件失败的原因就是服务器发送了X-Content-Type-Options: nosniff头。第二个最常见的原因是服务器发送了一个不在上面列表中的未知内容类型 。解决HTTP头的问题，要么修改服务器配置或者尝试在另外的服务器上托管.crx文件。

打包

本章描述如何给你的扩展打包。正如 综述 中提到的扩展文件是一个签名的ZIP文件，扩展名是crx。比如 myextension.crx.

注意: 如果你使用 Chrome Developer Dashboard,发布你的扩展，你将无需自己打包。你自己打包一个crx的唯一原因是你需要发布一个非公开版本，比如一个alpha测试版本给测试用户。

当你打包一个扩展到时候。这个扩展获得唯一的一对密钥，其中的公共密钥用于标识这个扩展，私密密钥用于保存私密信息和给这个扩展的各个版本签名。

创建一个包

为扩展打包的步骤:

1. 访问如下URL进入扩展管理页面:

   > **chrome://myextensions/extensions**

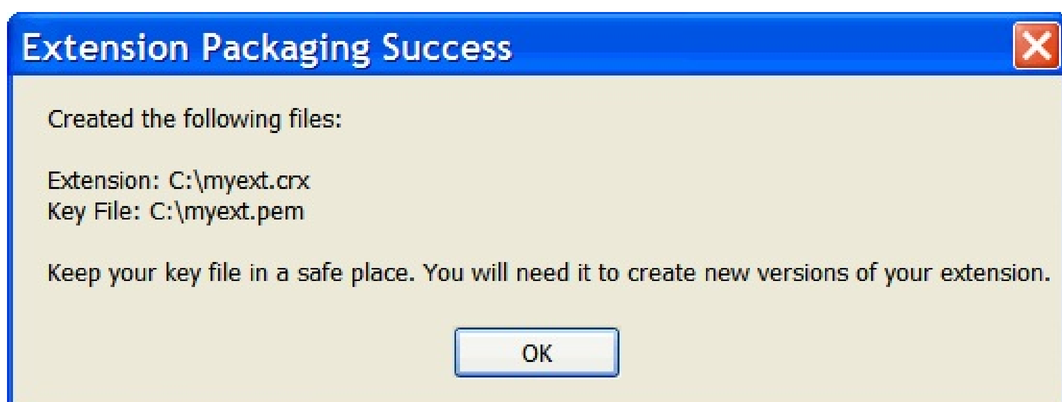
2. 如果开发人员模式边上有 + , 点击 + 以展开开发人员模式。

3. 点击"打包扩展程序"按钮，会出现一个对话框。
4. 在扩展程序根目录中填入你扩展所在的目录， 如： c:myext. (你可以忽略其他项，第一次打包时你无需指定私钥。)
5. 点击确定按钮。会生成两个文件： a .crx , 是一个真正的扩展文件，可以被安装。另一个 a .pem 文件, 是你的私钥文件。

请妥善保存你的私钥文件，尽可能放在安全的地方。在做以下事情的时候，你将需要用到它：

- 更新 这个扩展
- 使用 Chrome Developer Dashboard 上传这个扩展

如果扩展打包成功，你将看到如下的对话框，告诉你在哪里可以找到crx文件和pem文件:

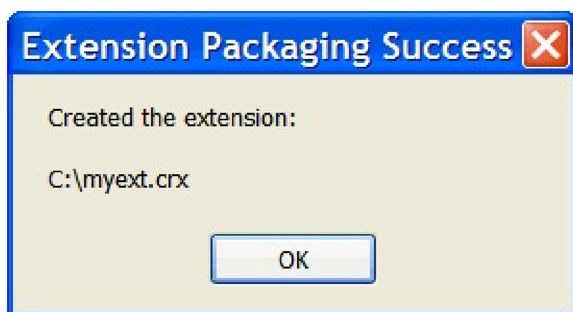


更新一个包

为你的扩展创建一个更新版本的步骤如下:

1. 增加 manifest.json文件中的版本号字段。
2. 访问如下URL进入扩展管理页面: **chrome://myextensions/extensions**
3. 点击"打包扩展程序"按钮，会出现一个对话框。
4. 在扩展程序根目录中填入你扩展所在的目录，如： c:myext.
5. 在私有密钥文件中填入你私有密钥所在的位置，如： c:myext.pem.
6. 点击确定按钮。

如果你更新扩展成功，你会看到如下对话框：



用命令行打包

另一种打包方式是用特定的命令行参数，--pack-extension指定扩展所在的文件夹，--pack-extension-key指定私钥所在的文件位置，然后调用 chrome.exe 下面是示例：

```
chrome.exe --pack-extension=c:myext --pack-extension-key=c:myext.pem
```

如果你不想看到对话框，请使用 --no-message-box。

包格式和脚本

获取更多创建crx文件的包格式的信息和脚本的要点，请参见：CRX 包格式。

规范和协议

应用设计规范

应用设计规范文档旨在帮助您迅速了解360极速浏览器应用开发的基本用户体验规范，作为您在设计产品时的参考，同时也有助于您的产品更快地通过我们的应用审核。 这份文档将不断丰富，目前我们先将360极速浏览器应用的一些基本用户体验原则阐述如下：

1. 收敛需求。将最核心的功能及服务呈现给用户。

1.1 一般说来，一个应用应聚焦于一项核心功能或服务，在进行界面外观设计、内容排版、功能导航、交互操作设计时也应围绕这个目标来进行；

1.2 避免在有限的界面上堆砌与本应用无直接关系的其它内容，包括无关的推广链接、导航、图片和无关的脚本、flash等内容；

2. 优化性能。让它运行起来更流畅、轻快。

2.1 保持应用的快速、稳定的访问非常重要，这将决定用户是否能长期使用您的应用，如果您的web服务需要临时停机维护，应当提前通过应用通知您的用户；

2.2 应用在后台运行或在网页上插入脚本运行时，尽量将对性能的影响降到最低。用户可能同时安装了大量应用，任何轻微影响性能的代码累加起来都会造成性能的大幅下降；

2.3 用户在使用一款应用时的预期与网页不同，他们需要更为流畅、自然的体验，建议在应用的页面切换以及数据加载尽量保持平滑的体验；

3. 优化交互体验。使用基于直觉的设计，让用户更容易操作。

3.1 避免直接将某个网站的现有网页简单嵌套进来，而不顾用户的使用体验。开发者应当考虑根据这个应用的需求来调整原有网页的形式（包括页面大小、图片大小、导航元素等等），或设计全新的UI模板。

3.2 在UI设计上，应用内各页面也应保持风格一致，并具有清晰的导航，让用户清晰地知道自己所处的位置；避免前后反差过大，使用户无所适从；

3.3 在交互设计上，用户的任何一个操作都应当获得清晰的反馈，比如点击播放一首歌曲（或者播放一段视频、打开一张较大的图片）时，缓冲加载过程中应该给用户展示清晰的反馈信息（如动态进度条或转轮动画等），让用户明确知道自己需要稍作等待和值得等待，而不是毫无反应，让用户误以为应用已经卡住了或点击无效； 为用户的操作提供指引，避免用户进入到某个状态时，无所适从。

4. 优化界面。提供精心设计的，优秀、人性化的应用界面。

4.1 为每个应用提供精美的图标，界面上使用精美的图片。优秀、人性化的应用界面能吸引用户使用，提高使用频率。让用户在使用过程中感到愉悦、亲切。

5. 不干扰用户的正常使用。

5.1 避免频繁打扰或在不恰当的时刻打扰用户，仅在用户可能需要时，用柔和的方式提醒用户。打扰用户的方式包括：自动播放声音、弹出窗口、对用户正在浏览的页面做出较大干扰等。

5.2 不干扰浏览器本身的功能。应避免应用所提供的功能，对浏览器本身的功能造成负面影响。

5.3 不干扰其他应用，开发者在设计应用时，应考虑到用户使用环境的复杂性，尽量避免与其他应用造成冲突，导致网页功能不正常，浏览器、应用不稳定等状况。

6. 在应用中不应插入过多广告，尤其不能让广告喧宾夺主造成对用户体验的伤害。

6.1 应用首页不应出现广告内容；

6.2 应用内页如含广告内容，须严格保证广告内容不影响、不干扰用户正常使用本应用；

6.3 应用内禁止出现弹出窗口、浮层或漂浮icon等类型的干扰型广告；

6.4 应用禁止各类恶意诱导用户点击广告的行为；

7. 应用应持续更新改进，并通过适当的方式告知用户您的改进；

7.1 好的应用是运营出来的，收集用户的意见建议并不断改进和更新，才能让您的应用更受欢迎；

7.2 如果您的应用已经通过网页实现了更新，建议及时在页面告知用户更新内容；

7.3 您可以在应用内合理的位置设立用户反馈的入口，使用户能将自己遇到的问题及时反馈给您，作为您改进应用功能和体验的参考依据；

开发人员协议

请务必认真阅读和理解本《360极速浏览器应用开放平台开发人员协议》（以下简称"协议"）中规定的所有权利和限制。在您参与360极速浏览器应用开放平台并使用应用中心分发产品时，您首先应接受本《协议》条款，如果您不接受本协议或违反了协议，奇虎360有权采取以下措施：驳回应用审核、下线产品、删除帐户、拒绝访问等。 本《协议》是开发人员与北京奇虎科技有限公司（下称"奇虎360"）之间关于利用360极速浏览器应用开放平台开发、发布、传播、发行的法律协议。

1. 定义

1.1 开发人员：指的是接受并依据本协议条款开发应用产品的任何公司、单位、个人及其他组织。

1.2 应用开放平台及审核：360应用开放平台网站由奇虎360运营，开发人员可在应用开放平台中上传应用产品。奇虎360将对上传的应用产品进行审核。

1.3 应用中心：360极速浏览器应用中心网站由奇虎360运营，开发人员可以将通过审核的应用产品将发布到应用中心网站，供360极速浏览器用户选择使用。

1.4 应用产品：指的是与360极速浏览器结合使用的、通过应用中心分发的软件、内容和数字材料。

2. 使用许可

2.1 开发人员使用360应用平台及应用中心应遵守本协议的约定以及法律、法规之相关规定。

2.2 开发人员使用应用中心分发产品时，应充分保护用户的隐私及其他合法权利。遵循软件行为不越界、用户隐私不上传、操作必须明示用户的原则。

2.3 开发人员应对在应用中心中发布的任何产品以及操作所引发的后果承担全部责任（包括?奇虎360?或第三方可能因此遭受的损失或损害），奇虎360 对开发人员或任何第三方不承担任何责任。

2.4 开发人员向应用开放平台上传产品应向用户提供必要的产品信息及使用说明。未正确上传的产品将不会在应用中心中发布。

2.5 开发人员随时可以要求从应用中心中删除产品，但不应影响已经下载了产品的用户的正常的使用与许可权，开发人员有删除应用的需求，可发邮件至app-c@360.cn与我们联系。

2.6 奇虎360 有权审查或测试开发人员发布的产品或产品的源代码，奇虎360 保留自行决定拒绝将产品加入应用中心的权利。

2.7 开发人员应保证向奇虎360所提供的信息准确性和及时性。作为产品规范的一部分，奇虎360 可能会要求开发人员将个人信息（例如姓名和电子邮件地址）加入产品信息的文件中。并且奇虎360可以在产品目录中或产品展示时使用该信息。

2.8 开发人员发布的应用产品中存在以下情形的，奇虎360 可以自行决定不在应用中心中提供该产品、要求从应用中心中删除该产品、以远程方式从用户的系统或设备中停用或删除该产品、或举报、过滤、修改相关材料（包括但不限于说明、屏幕截图或元数据）或重新分类该产品。

2.8.1 违反国家法律、法规强制性规定及本协议约定的；

2.8.2 侵犯了任何第三方的知识产权或任何其他权利的；

2.8.3 发布或分发方式不正确；

2.8.4 奇虎360 认为其携带病毒或将其视为恶意软件、间谍软件或会对奇虎360或第三方的网络产生不利影响；

2.8.5 可能会导致 奇虎360 或任何第三方需要承担责任；

2.8.6 产品的运行影响了奇虎360服务器的性能；

2.8.7其他违反了奇虎360的托管政策或服务条款的。

2.8.4 奇虎360 认为其携带病毒或将其视为恶意软件、间谍软件或会对奇虎360或第三方的网络产生不利影响；

2.9开发人员对产品的更新应遵守与产品发布相同的条款。

3. 产品评分

3.1 应用中心将会开放用户对开发人员发布的应用产品进行评分，并依据产品评分并综合考虑其他因素来确定应用产品在应用中心中的展示位置。更改产品展示位置的权利归 奇虎360 独家所有，奇虎360保留自行决定如何向用户提供产品的权利。

3.2 开发人员对于用户评分有异议的，可以与发邮件到app-c@360.cn与我们联系。

4. 权利限制

4.1 开放平台上传的应用不得包含以下行为（包括产品或其他材料的开发或发布）：

4.1.1 违反法律、法规的强制性规定；

4.1.2 干扰、修改、破坏、损坏或以未经授权的方式访问任何第三方（包括但不限于360极速浏览器用户、奇虎360 或任何互联网信息服务提供者）的计算机、硬件、设备、服务器、网络、数据或其他财产或服务；

4.1.3 故意传播计算机病毒等破坏性程序及其他任何危害计算机信息网络安全的；

4.1.4 未经允许，对计算机信息网络中存储、处理或者传输的数据和应用程序进行删除、修改或者增加；

4.1. 5 侵犯他人的知识产权及其他合法权益；

4.1.6 制造干扰信息、恶意信息、低俗信息影响用户体验；

4.1.7 利用应用产品制作、复制、发布、传播含有国家法律、法规禁止的内容的信息，及通过链接展示上述信息。

4.2 开发人员不得以任何方式诱导用户到模仿或假冒应用中心的任何其他网站。

4.3 未经奇虎360的书面许可，应用产品中不得使用或包含 NPAPI 插件。

4.4 保留权利：本协议未明示授权的其他一切权利仍归奇虎360所有， 开发人员使用其他权利时必须获得奇虎360的书面同意。

5. 权利授予

5.1 开发人员在此授予奇虎360及其关联企业非专有的、全球范围的免版权费许可，允许奇虎360 及其关联企业复制、存储、传播、链接至、翻译、公开展示、测试、分发或以其他方式使用开发人员在应用开放平台上传的产品、 产品中包含的、 产品所访问的或通过产品传播的任何内容，具体情况根据您在而定。

5.2 开发人员在此向用户授予非专有的、全球范围的永久许可，允许用户在360极速浏览器中下载、安装和使用开发人员在应用开放平台上传的产品、 产品中包含的、 产品所访问的或通过产品传播的任何内容。

5.3 开发人员可以在自己的产品中添加一份单独的最终用户许可协议 (EULA)，取代前款对用户的许可授权。

5.4 开发人员声明并保证有权向奇虎360及其关联企业以及最终用户授予以下产品和内容的相应许可：开发人员在应用开放平台上传的产品、 产品中包含的、 产品所访问的或通过产品传播的任何内容，而且将努力保持拥有这些权利。

5.5 除本协议中授予的许可权利外，开发人员保留自己在产品中享受的所有权利；对这些权利的保护和行使负有责任。

5.6 开发人员授权奇虎360 及其关联企业在本协议的有效期内有限的、非专有的许可，允许 奇虎360 及其关联企业为了随应用中心一起使用以及为了履行本协议规定的义务而展示开发人员的品牌特征。包括但不限于商号、商品名、商标、服务标记、标识、域名和其他显著的品牌特征。

5.7 开发人员授权奇虎360 及其关联企业在推广360极速浏览器应用中心、 奇虎360 产品和服务时使用应用产品、 产品中包含的、您产品所访问的或通过您产品传播的任何内容的开发人员品牌特征、屏幕截图、视频和演示。包括但不限于：演示文稿、营销材料、营销活动、开发人员活动、财务报告、网站列表、新闻稿和客户列表。

6. 隐私权保护

6.1 为了更好地改进应用中心和服务，在用户加入《用户体验改进计划》后，奇虎360可能会从应用中心和用户计算机、设备或其他硬件中收集某些使用情况统计信息，包括但不限于，有关应用中心和产品的使用方式的信息。奇虎360不会将此数据与用户的个人身份信息相关联。

6.2 奇虎360 会对收集的数据进行汇总分析，以便为用户和开发人员提供更好的应用中心体验：开发人员出于改进产品的需要可以向奇虎360 提交书面申请，奇虎360会根据申请提供一定的汇总数据。

7. 免责与责任限制

7.1使用应用中心的风险由开发人员自行承担，在适用法律允许的最大范围内，对因使用或不能使用应用中心所产生的损害及风险，包括但不限于直接或间接的个人损害、商业赢利的丧失、贸易中断、商业信息的丢失或任何其它经济损失，奇虎360不承担任何责任。

7.2对于因电信系统或互联网网络故障、计算机故障或病毒、信息损坏或丢失、计算机系统问题或其它任何不可抗力原因而产生损失，奇虎360不承担任何责任。

7.3奇虎360 不提供任何类型（明示或隐含）的担保和条件，包括但不限于对适销性、特定用途适用性，以及不侵害他人权利的隐含担保和条件。

7.4 开发人员违反本协议规定，对奇虎360造成损害的。奇虎360有权采取包括但不限于下线产品、删除账户、拒绝访问、法律追究等措施。

7.5 因以下由于开发人员责任引起或产生的任何争议、索赔、诉讼，开发人员应对奇虎360及其关联企业造成的全部损失予以赔偿，并有责任在法律允许的最大范围内对奇虎360及其关联企业进行辩护并使其免受损害。

7.5.1开发人员的产品使用的行为违反了本协议及适用法律法规，

7.5.2开发人员的产品侵犯了他人的商标、商业机密、专利、外观设计或其他知识产权，或侵犯他人名誉权、隐私权的

7.5.3因产品或扩展程序使用行为而引起的任何第三方索赔。

8. 法律及争议解决

8.1 本协议适用中华人民共和国法律。

8.2 因本协议引起的或与本协议有关的任何争议，各方应友好协商解决；协商不成的，任何一方均可将有关争议提交至北京仲裁委员会并按照其届时有效的仲裁规则仲裁；仲裁裁决是终局的，对各方均有约束力。

9. 其他条款

9.1 如果本协议中的任何条款无论因何种原因完全或部分无效或不具有执行力，或违反任何适用的法律，则该条款被视为删除，但本协议的其余条款仍应有效并且有约束力。

9.2 奇虎有权根据有关法律、法规的变化以及公司经营状况和经营策略的调整等修改本协议。修改后的协议会在奇虎360应用开放平台网站（网址是 ：http://open.chrome.360.cn/）上公布。当发生有关争议时，以最新的协议文本为准。如果不同意改动的内容，开发人员如果继续使用应用开放平台或应用中心服务，则视为您接受本协议的变动。

9.3 本协议的一切解释权与修改权归奇虎360。

免责声明

此页面的开发文档由360工程师翻译自Chromium官方的扩展开发文档，您可以随意传播、转阅。